

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Fidelização e Ofertas em Realidade Aumentada

José Tadeu Pinto Marques



Mestrado Integrado em Engenharia Informática e Computação

Orientador: António Miguel Pontes Pimenta Monteiro

Co-orientador: Jorge Alves da Silva

16 de Julho de 2014

Fidelização e Ofertas em Realidade Aumentada

José Tadeu Pinto Marques

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo Júri:

Presidente: Maria Cristina de Carvalho Alves Ribeiro

Arguente: Paulo Miguel de Jesus Dias

Vogal: António Miguel Pontes Pimenta Monteiro

16 de Julho de 2014

Resumo

O privilégio de estudar e prosseguir um ciclo de estudos, que apela à investigação, e as oportunidades de investigação que hoje em dia existem levam a uma cada vez maior evolução tecnológica, visível no dia-a-dia. Dos anos mais recentes, até agora, surgiram os *smartphones* que hoje em dia fazem parte do quotidiano de uma grande parte da população a nível Mundial. Torna-se portanto importante aproveitar este recurso para o desenvolver, e ainda mais para uma empresa que pretenda inserir-se fortemente no mercado.

Da mesma forma que os *smartphones* cresceram nos últimos anos, está agora em emersão a realidade aumentada. Esta é recente, ainda não muito explorada, e certamente passará por aqui o futuro da sociedade. Então torna-se fulcral perceber onde podemos intervir, nessa área, por forma a não perder o ritmo a que a comunidade se desenvolve.

É também importante denotar a importância da plataforma Android nos dias de hoje, que, juntamente com o iOS, está presente a nível global em todos os continentes e países, tendo um grande impacto inclusive no quotidiano de todas as sociedades, assumindo assim um papel relevante no contexto desta tese.

Face a esta realidade, é importante para uma empresa que trabalha com cupões de desconto e fidelidade manter-se inovadora e por isso, investir nestas 2 grandes áreas.

Assim, neste relatório é abordada uma forma de conseguir trazer propostas de fidelização através da realidade aumentada até aos *smartphones* dos diversos utilizadores de aplicações que gerem essa fidelização.

Para tal, serão estudados os temas de posicionamento em 2 dimensões, tanto a nível de *indoor* como de *outdoor*, bem como de 3 dimensões, sendo que para isso se vai falar de como medir a altitude a que um utilizador se encontra relativamente ao nível do mar.

A orientação será também estudada, para se saber para onde está o utilizador virado, por forma a saber onde são apresentadas as informações que se pretende exibir no ecrã do *smartphone*.

Para a realização do proposto, é preciso trabalhar com o conceito de realidade aumentada. Então, são apresentadas algumas tecnologias pertinentes para o trabalho, bem como algumas aplicações já existentes com funcionalidades parecidas e interessantes para o tema proposto neste relatório.

De seguida, é apresentada de forma sucinta uma proposta de solução. Isto é, de acordo com o estudado no estado de arte, falar-se-á do que foi utilizado em cada um dos tópicos de interesse.

Por fim, são expostos alguns testes realizados com a aplicação desenvolvida e de seguida é elaborada uma conclusão sobre o trabalho realizado.

Abstract

The privilege of studying and pursuing a course of studies, that incentivates investigation, and the research opportunities that now exist lead to a continuously increasing in technology development, visible every day. From the past recent years, to the present, smartphones emerged and nowadays are part of the everyday life of most of the World's population. It is therefore important to take advantage of this feature to develop it, and even more for a company to be able to insert themselves into the market and compete on it.

The same way that smartphones have grown in recent years, it is now emerging the augmented reality. This is recent, and still not much explored, and certainly will have an important role in the society's future. It then becomes crucial to realize where we can intervene in this area, so as not to lose the momentum that the community develops.

Given this reality, it is important for a company that works with discount coupons and loyalty to remain innovative and therefore invest in these two major areas.

Thus, in this report, it is addressed one way to show new discount coupons through augmented reality to the smartphones of as many people as possible.

To do this, one studied 2 dimension positioning systems, both in indoor and outdoor. 3 dimension situations are also studied, and for that the altitude that an user is in comparison to sea level will be measured.

The orientation is also studied, to know where is the user facing in order to know where the information, that you want to show in the screen, is shown.

To perform the proposed work, it is essential to work with the augmented reality concept. Then, it will be presented some useful technologies to the work, as well as some already existing applications with similar functionalities and interesting for the proposed theme in this report.

Next, a solution is succinctly presented. According to the study made in the state of art, it is shared what is going to be used in each topic.

Finally, some tests are exposed related with the developed application and then a conclusion about the work done is also made.

Conteúdo

1	Introdução	1
1.1	Enquadramento	1
1.2	Problema	2
1.3	Motivação	2
1.4	Objetivos	3
1.5	Benefícios esperados	3
1.6	Estrutura da Dissertação	3
2	Estado da Arte	5
2.1	Posicionamento e Orientação	5
2.1.1	Localização em espaços indoor	6
2.1.1.1	GPS	6
2.1.1.2	Proximidade	6
2.1.1.3	Trilateração	7
2.1.1.4	RFID	7
2.1.1.5	Fingerprinting	7
2.1.1.6	ToA,TDOA	8
2.1.1.7	AOA	8
2.1.1.8	Dead Reckoning	8
2.1.1.9	Map Matching	9
2.1.2	Localização em espaços outdoor	9
2.1.2.1	GPS	9
2.1.2.2	A-GPS	9
2.1.2.3	Wi-fi	9
2.1.3	Google Location Services	10
2.1.4	Orientação	10
2.1.5	Altitude	11
2.2	Câmara/Realidade Aumentada	14
2.2.1	Abordagens	14
2.2.1.1	QR-Code	14
2.2.1.2	Reconhecimento Logotipo	15
2.2.2	Tecnologias	15
2.2.2.1	AppCelerator Titanium	15
2.2.2.2	Utilização direta do Android	16
2.3	Trabalhos Relacionados	17
2.3.1	Layar	17
2.3.2	Junaio + Valpak	17
2.4	Conclusões	18

CONTEÚDO

3	Arquitetura	21
3.1	Requisitos e Funcionalidades	21
3.2	Perspetiva Geral	22
3.3	Servidor e Aplicação	23
3.4	Módulo de Realidade Aumentada	23
3.5	Fluxo do Módulo	24
3.6	Sequência de Operações	26
4	Implementação	29
4.1	Tecnologias	29
4.2	Realização	29
4.2.1	Módulo de Realidade Aumentada	30
4.2.1.1	Estrutura Geral	30
4.2.1.2	Localização	30
4.2.1.3	Sensores	31
4.2.1.4	Informação das lojas	33
4.2.1.5	Posicionamento no ecrã	33
4.2.1.6	Overlay de imagens	39
4.2.2	Aplicação Cardmobili	39
4.2.3	Servidor	40
5	Resultados	41
5.1	Primeira versão de teste	41
5.2	Versão integrada na aplicação <i>Cardmobili</i>	42
6	Conclusões e trabalhos futuros	47
6.1	Resumo e conclusões	47
6.2	Satisfação dos objetivos	48
6.3	Desenvolvimentos necessários	48
6.4	Trabalho Futuro	48
	Referências	51

Lista de Figuras

2.1	Precisão de altitude através de GPS [GL]	11
2.2	Precisão de altitude através de sensor de pressão [JE10]	12
2.3	Imagem ilustrativa de um exemplo de QR-Code ¹	14
2.4	Logótipo do AppCelerator Titanium	15
2.5	Imagem do Logótipo do Layar ²	17
2.6	Imagem dos Logótipos da Junaio e Valpak ³⁴	17
2.7	Imagem elucidativa da interface da aplicação da <i>Junaio</i>	18
2.8	Imagem representativa de uma tabela com os diversos algoritmos indoor estudados [FNI13]	19
2.9	Imagem representativa de uma tabela com os diversos algoritmos outdoor estudados [FNI13]	20
3.1	Esquema base de interação entre as diversas componentes da aplicação.	23
3.2	Imagem ilustrativa da interface esperada da aplicação	25
3.3	Imagem ilustrativa da interface esperada da aplicação	26
3.4	Imagem ilustrativa do diagrama de sequência esperado para o módulo	27
4.1	Comparação entre os antigos processos de cálculo da localização e os mais recentes existentes na última versão do <i>Google Location Services</i> ²	31
4.2	Representação dos 3 eixos num dispositivo móvel [Dab11]	31
4.3	Ilustração representativa do eixo <i>azimuth</i> [Dab11]	32
4.4	Ilustração representativa do eixo <i>pitch</i> [Dab11]	32
4.5	Ilustração representativa do eixo <i>roll</i> [Dab11]	33
4.6	1) Situação em que o campo de visão ultrapassa o valor 360 ou tem valores negativos	35
4.7	2) Situação em que o campo de visão não ultrapassa o valor 360 ou tem valores negativos	35
4.8	a) Ângulo da loja maior que 0 e menor que limite direito do campo de visão . . .	37
4.9	b) Ângulo da loja maior que 0 e limite direito do campo de visão menor que ângulo da loja	38
5.1	Primeira versão do protótipo	42
5.2	Aplicação <i>Cardmobili</i> com módulo de realidade aumentada integrado	43
5.3	Versão final do protótipo	43
5.4	Versão final do protótipo	44

¹http://pt.wikipedia.org/wiki/C%C3%B3digo_QR

²<https://www.layar.com/>

³<http://www.junaio.com/>

⁴<http://www.valpak.com/coupons/home>

LISTA DE FIGURAS

5.5	Versão final do protótipo	44
5.6	Exibição da informação da loja	44
5.7	Exibição da informação da loja	45
5.8	Exemplo de cartao do utilizador	45

Abreviaturas e Símbolos

A-GPS	Assisted Global Positioning System
GPS	Global Positioning System
Indoor Positioning	Sistema que retorna uma posição precisa dentro de estruturas fechadas
GSM	Global System for Mobile Communications
Outdoor Positioning	Sistema que retorna uma posição precisa em ambientes abertos (sem obstáculos entre o dispositivo e os satélites)
PA	Pontos de Acesso
QR-Code	Quick Response code
RA	Realidade Aumentada
RFID	Radio-Frequency IDentification
RSS	Received Signal Strength
SMS	Short Message Service
VOR	VHF Omnidirectional Ranging
Wi-Fi	Uma facilidade que permite computadores, <i>smartphones</i> ou outros dispositivos conectarem-se à internet ou comunicarem entre si sem fios dentro de uma determinada área
WWW	<i>World Wide Web</i>

Capítulo 1

Introdução

O desenvolvimento da tecnologia nas últimas décadas permitiu a entrada de novos conceitos no quotidiano da sociedade, proporcionados pela utilização em massa dos *smartphones* e outros dispositivos dotados de *GPS* (Global Positioning System). Outros conceitos como a *realidade aumentada* (RA - tecnologia que integra elementos virtuais com elementos do mundo real, podendo ou não permitir a interação entre eles) ainda estão em crescimento, não tendo ainda um papel tão importante. Contudo, com o crescimento tecnológico e a investigação permanente, é espectável que o futuro passe, de forma acentuada, pela realidade aumentada.

1.1 Enquadramento

É neste contexto que se insere a empresa *CardMobili* ¹, que trabalha a área das carteiras digitais e em serviços e processos de fidelização de clientes, principalmente através da utilização de cartões de fidelização ou cupões (por exemplo vales de desconto) e assume como palavra chave a inovação.

A Carmobili desenvolveu a tecnologia base de diversas carteiras digitais implementadas pelo Mundo. Algumas das suas participações são por exemplo a *Meo Cardmobili* ou a *Meo Wallet*, ambas carteiras digitais, em que a segunda oferece também a possibilidade de fazer pagamentos nas lojas que suporta (contudo, é uma aplicação até ao momento ainda em testes).

Segundo a *CardMobili*, "todos os anos mais de um terço dos cupões de descontos distribuídos não são rebatidos. Uma das principais causas para este problema é a dificuldade na descoberta e acesso a estes coupons pelos utilizadores". Perante esta realidade, a empresa acredita que tecnologias como a *realidade aumentada* podem facilitar o acesso a estes cupões por parte dos utilizadores, e assim, desta forma, levar à diminuição da percentagem de cupões não rebatidos. Sistemas como o *GPS* ou o *Wi-Fi* (entre outros) assumem especial relevância neste contexto, uma vez que permitirão identificar a posição em que se encontra um utilizador dispondo de um dispositivo móvel, em ambientes *Outdoor* ou *Indoor*, levando facilmente à criação de aplicações que

¹ www.cardmobili.com

tiram partido desse conhecimento(*location-aware*). Em ambientes *Indoor* a aquisição de posicionamento é mais complexa, levando a um estudo mais detalhado sobre a melhor maneira de o conseguir, estudando diversos algoritmos e métodos passíveis de serem aplicados para no fim se eleger o melhor e se proceder à sua implementação.

1.2 Problema

Num processo de fidelização de clientes em que uma empresa usa cupões de desconto para incentivar os seus clientes a voltarem, o problema que se destaca é a quantidade de cupões de desconto que todos os anos não são rebatidos e que a *Cardmobili* estima serem cerca de 1/3 (um terço) dos que são produzidos. Muitas das vezes a ocorrência deste problema está relacionada com o facto dos utilizadores não disporem desses cupões na altura ou se esquecerem que os possuem, sendo portanto útil dar aos potenciais utilizadores o conhecimento destas propostas na hora e locais oportunos. Outro motivo prende-se com o fato de que para alguns utilizadores o encontrar destes cupões pode ser um processo pouco direto, diminuindo o seu interesse e fazendo com que estes não os procurem. Também pode ser uma razão preponderante o fato de por vezes os utilizadores simplesmente não saberem onde se encontram determinadas lojas com descontos orientados para si. Ainda que saibam de alguns dos descontos, desconhecem alguns locais onde os poderiam rebater.

1.3 Motivação

Com o constante crescimento tecnológico a utilidade que dispositivos como *smartphones* podem ter no quotidiano está também a crescer continuamente. Desta forma, acompanhar este crescimento é essencial para qualquer empresa, como o é a *CardMobili*. Será útil conseguir aproveitar todos os recursos possíveis com a finalidade de conseguir aumentar o número de cupões rebatidos todos os anos.

Com a crescente capacidade gráfica e computacional dos *smartphones*, é possível a construção de aplicações práticas utilizando a realidade aumentada que também tem conhecido um crescimento já notado. Começar desde já a apostar nesta área será uma mais valia em termos de negócio uma vez que é uma área ainda pouco conhecida do público mas cuja importância começa aos poucos a aumentar. Deste modo, o desenvolvimento antecipado nesta área poderá constituir uma vantagem para a empresa no futuro.

Muitas vezes o conhecimento da posição do utilizador é crucial para estas aplicações. A utilização do GPS para este efeito pode trazer dificuldades, principalmente quando o utilizador se encontra no interior de edifícios. Em ambiente *indoor*, há outras possibilidades de se atingir um resultado, aparentando assim ser interessante investigar dentro dessa área a fim de descobrir a melhor implementação ou implementações. Esta representa uma questão extremamente importante uma vez que o trabalho que se pretende desenvolver, sendo *location-based*, prevê-se que funcione em qualquer lugar, seja em espaço aberto ou não.

Posto isto, entende-se que a realização desta dissertação seja útil para a empresa bem como para a integração no mercado de um conceito ainda não muito explorado de ofertas de cupões de fidelização e desconto através da realidade aumentada em aplicações móveis para facilmente identificar lojas e locais de utilização desses instrumentos.

1.4 Objetivos

É possível referir como principal objetivo fornecer aos utilizadores da aplicação *CardMobili* a possibilidade de facilmente descobrirem que programas de fidelização, ofertas e comércio têm ao dispor em seu redor utilizando a câmara do seu dispositivo *smartphone*, criando assim um serviço inovador para os clientes e comerciantes associados. Uma vez que a empresa pretende integrar a aplicação de realidade na sua própria aplicação já existente, o protótipo que vai ser desenvolvido não será uma aplicação base mas sim um módulo para posterior integração com qualquer aplicação *Android* que o pretenda fazer. Na prática, subdividindo em tópicos, pretende-se desenvolver uma aplicação que deverá:

- saber a localização das lojas físicas dos comerciantes associados a programas de fidelização ou cartões que a Cardmobili disponibiliza para outras empresas;
- saber a localização e orientação atual do *smartphone* que está a correr a aplicação, seja em ambientes interiores ou exteriores;
- saber quais os cupões específicos do utilizador em cada loja;
- apresentar, através de realidade aumentada, no *smartphone* cupões de fidelização/desconto específicos de cada loja;
- testar com um dispositivo a aplicação e tirar conclusões sobre a sua utilização;

À medida que se forem resolvendo estes objetivos será também escrito o relatório final da dissertação.

1.5 Benefícios esperados

Uma vez concluídos os objetivos mencionados, espera-se que a empresa beneficie de uma solução funcional que mostre lojas e cupões através da realidade aumentada servindo de base para possíveis futuras apostas nesta área. Outro benefício do desenvolvimento desta tese será o estudo feito relativamente à precisão ao nível da altitude e à diferença de altitudes entre dois pontos distintos em altitudes diferentes.

1.6 Estrutura da Dissertação

Para além da introdução, esta dissertação contém mais 5 capítulos. No capítulo 2, é descrito o estado da arte, são apresentados trabalhos relacionados e descritas tecnologias de interesse para o desenvolvimento da aplicação. No capítulo 3, é descrita a arquitetura geral do sistema sem especificar detalhes de implementação. Esses serão abordados no capítulo seguinte, o capítulo 4.

Introdução

No capítulo 5 serão ilustrados alguns exemplos de como evoluiu e como funciona a aplicação aquando da conclusão do período de desenvolvimento da mesma. Também serão comentados os resultados do trabalho.

Por fim, no capítulo 6 será feito um resumo e análise do trabalho e estudo efetuados, falando do que foi feito, do que ficou por fazer e de possíveis melhorias que poderão vir a ser implementadas no futuro.

Capítulo 2

Estado da Arte

De modo a poder-se desenhar uma solução satisfatória para o problema descrito e alcançar os objetivos definidos, temos de estudar um conjunto de técnicas que se nos afiguram poderem ser as mais adequadas. Serão estudados os temas mais importantes para o objetivo desta dissertação como:

- A obtenção fiável do posicionamento do utilizador, que será descrito na secção 2.1 ;
- A aquisição de imagem real e a sua combinação com elementos gráficos sintetizados, onde se falará da sua utilização no trabalho e possibilidades de integração com a aplicação sugerida. Esta integração constituirá uma aplicação da realidade aumentada, também introduzida na secção 2.2;

Por fim, serão identificados alguns trabalhos que estão relacionados com o tema desta dissertação e serão tiradas algumas conclusões pessoais sobre o material já existente sobre este tema.

2.1 Posicionamento e Orientação

Como foi descrito na introdução [1], o conhecimento da posição do utilizador tem um papel preponderante para que se atinjam os objetivos propostos. A exibição da informação pretendida no local correto implica o conhecimento preciso da posição do utilizador da aplicação e qual a sua orientação, também é necessário conhecer com precisão onde estão localizadas as lojas e quais são estas.

Em dispositivos móveis existem várias técnicas de localização com diferentes capacidades de precisão, custo de memória ou custo de energia por exemplo.

Estes resultados podem também ser diferentes conforme as técnicas estejam a ser usadas em meios *indoor* ou *outdoor* uma vez que na primeira situação os sinais usados para determinar a posição do utilizador podem ser muito enfraquecidos devido à existência de paredes, tetos ou

outras estruturas ¹ e na segunda situação os algoritmos utilizados também não são 100% precisos, apresentando diferentes capacidades de precisão.

Assim, para determinar a posição será necessário recorrer a diversas tecnologias dos dispositivos móveis, como GPS, Wi-Fi, câmara e outros sensores como acelerómetro, giroscópio ou sensor de pressão.

2.1.1 Localização em espaços indoor

O posicionamento também pode ter de ser feito em meios *indoor* como por exemplo centros-comerciais, que serão importantes para os objetivos propostos para a dissertação. Assim, é importante estudar os diferentes algoritmos existentes para resolver este tipo de localização.

Quando é necessário determinar este posicionamento, uma das alternativas que existe envolve a tecnologia WLAN (*wireless lan*). Esta tecnologia não foi desenhada com o intuito de determinar a posição de um dispositivo. Contudo, medições de força de sinal do sinal transmitido, seja por pontos de acesso (PA) ou por estações, permitem deduzir a localização de qualquer utilizador de dispositivos móveis. [LSDR06].

2.1.1.1 GPS

"O sistema de posicionamento global (do inglês global positioning system, GPS) é um sistema de navegação por satélite que fornece a um aparelho recetor móvel a sua posição, assim como informação horária, sobre todas condições atmosféricas, a qualquer momento e em qualquer lugar na Terra, desde que o receptor se encontre no campo de visão de pelo menos quatro satélites GPS." ²

Existem vários fatores que condicionam os resultados do GPS em termos de precisão. De acordo com Kourogi, Sakata, Okuma e Kurata [KSOK06], 3 dos mais importantes são:

- erro causado pelo efeito de multi-caminhos
- erro causado pelo atraso da propagação do sinal
- erro causado por o dispositivo que tem GPS estar a detetar 3 ou menos satélites

Ora uma vez que as paredes levam ao aparecimento de multi-caminho e permitem o atraso de propagação de sinal, conclui-se facilmente que a funcionalidade de GPS é uma fraca opção perante os objetivos que se pretendem atingir.

2.1.1.2 Proximidade

A proximidade é um dos métodos de posicionamento mais simples de implementar [FNI13]. Consiste em associar a localização do utilizador à localização de um ponto de referência.

A precisão destes algoritmos depende da distância a que um ponto de referência pode ser detetado (típicamente será um valor entre 100 metros e 10 quilómetros em redes de GSM) [RD08].

¹http://en.wikipedia.org/wiki/Indoor_positioning_system/

²http://pt.wikipedia.org/wiki/Sistema_de_posicionamento_global/

2.1.1.3 Trilateração

O algoritmo de trilateração também é relativamente simples. São necessárias 3 estações base (ou mais) com coordenadas conhecidas. Se uma distância R do PA ao utilizador for mensurável, é traçado um círculo com raio R . Fazendo o mesmo processo para cada um dos 3 ou mais PA, os círculos interseccionar-se-ão num ponto que será a posição do utilizador. Contudo, os resultados obtidos são forças de sinal em vez de distâncias, portanto devem ser apropriadamente convertidos para unidades de distância primeiro [LSDR06].

O problema deste método prende-se com o facto de ser afetado por multi-caminhos, paredes, tectos ou até mesmo outros sinais. Como tal, em interiores continua a ser um método com algumas lacunas.

2.1.1.4 RFID

Radio Frequency Identification (RFID) é um método de identificação automático. Um sistema RFID consiste numa tag, um leitor e uma antena. A tag é um *transponder* (aparelho que emite um sinal identificador em resposta a um sinal de interrogação recebido) que pode ser ligada a um produto, animal, ou pessoa, com o propósito de identificação usando ondas de rádio. O leitor (*transceiver*) é capaz de ler a informação guardada na tag quando em proximidade [GR08].

A técnica de posicionamento por RFID pode ser sub-dividida em 4 passos:

- instalar tags RFID em pontos específicos
- guardar informação de localização muito precisa juntamente com outras informações necessárias das tags
- adicionar um módulo leitor de RFID no sistema de navegação
- usar esta nova informação de localização

Para usar RFID como sistema de posicionamento, uma estratégia é instalar leitores RFID em certos pontos específicos, como entradas de edifícios ou lojas, para detetar um objeto quando passar por este [CJJA04]. De notar que os autores *Chon, Jun, Jung e An* admitem que:

"This technique, however, would not replace GPS rather it is a complementary technique" [CJJA04].

,ou seja,

"Esta técnica, contudo, não substituirá o GPS. Em vez disso será uma técnica complementar."

Esta constatação diz-nos que o RFID não é utilizável independentemente do GPS. Assim, não faz sentido estar a usar uma tecnologia que terá mais custos para o dispositivo móvel em vez de os diminuir.

2.1.1.5 Fingerprinting

De acordo com Jian [Jia12], recentemente, WLAN RSS Fingerprinting tem-se tornado uma das técnicas mais exploradas em localização *indoor*. É um método relativamente fácil de pôr

em execução e é tolerante ao ruído de sinal e, portanto, consegue assim atingir resultados mais precisos.

O processo consiste em 2 fases [SK08]:

- Treino *offline*. O RSS de diversos PAs, em diferentes pontos de uma grelha imaginária, representativa da área a ser estudada, são reunidos e guardados numa base de dados de *fingerprint* frequentemente chamada de *radio map*. O vetor dos valores médios dos RSS num determinado ponto na grelha de localização *fingerprint* desse ponto.
- Determinação *online*. O dispositivo online vai medir uma amostra de vetores *fingerprint* com os valores RSS de diferentes PAs nas suas posições. A amostra é enviada para um servidor centrar na infraestrutura WLAN. O servidor compara a *fingerprint* medida com as que tem na base de dados (*radio map*) para determinar a posição do dispositivo na grelha. Então, o resultado estimado é devolvido ao dispositivo.

A grande desvantagem deste algoritmo é ter de treinar o sistema com o maior número de leituras de sinais de rádio possível. Como estes sistemas apenas permitem que a fase *offline* se realize antes do seu verdadeiro uso, os sistemas só serão precisos enquanto a base de dados treinada ainda fizer sentido perante os dados reais, caso contrário, terá de voltar a ser treinada. [Bol11].

2.1.1.6 ToA,TDOA

"Time of Arrival" é a quantidade de tempo que um sinal demora a percorrer o caminho entre transmissor e receptor. Como a frequência de propagação do sinal é constante e conhecida, o tempo de viagem de um sinal pode ser usado para calcular diretamente a distância. Medições múltiplas podem ser combinadas com a Trilateration referida em 2.1.1.3. Esta é a técnica usada pelo GPS e, como tal, sofre dos mesmo problemas de obstáculos de paredes ou tetos, e refração e difração do sinal em objetos tornando esta técnica muito pouco precisa em meios interiores³.

2.1.1.7 AOA

É semelhante à técnica *Trilateration* referida em 2.1.1.3, com a diferença que os dados são medidos pelos ângulos formados entre os PA em vez das distâncias. Os problemas deste método são semelhantes também aos da *Trilateration* [RD08].

2.1.1.8 Dead Reckoning

O *dead reckoning* estima a atual posição do utilizador baseando-se na última posição determinada e incrementando essa posição baseado na velocidade conhecida ou estimada em função do tempo decorrido. Uma desvantagem deste método é que a falta de precisão é cumulativa, portanto o desvio da posição fixa aumenta com o tempo. A razão prende-se com o facto de que novas posições são calculadas inteiramente a partir de posições anteriores [FNI13].

³http://en.wikipedia.org/wiki/Indoor_positioning_system

2.1.1.9 Map Matching

Map Matching é o processo de alinhar uma sequência de posições do utilizador observadas, com a rede da internet, num mapa digital para suportar módulos de navegação de sistemas de transporte inteligentes que requerem informação de localização. [LZZ⁺09].

O propósito desta técnica é identificar o segmento de estrada correto que o veículo se encontra a percorrer e determinar a sua localização e atribuí-la a um local específico no dado mapa. Contudo, em algumas situações é difícil obter uma precisão considerável e desta forma há alguns algoritmos de map matching que permitem uma aproximação mais correta da posição real.

2.1.2 Localização em espaços outdoor

Serão agora abordadas algumas das técnicas mais conhecidas para posicionamento em meios exteriores, explicando em alguns casos o seu funcionamento e falando de forma sucinta das suas vantagens/desvantagens.

2.1.2.1 GPS

O GPS é uma tecnologia que está presente em quase todos os *smartphones* de hoje em dia. Como tal, a sua utilização permite que mais utilizadores tenham acesso às tecnologias que de si são dependentes. Segundo Reyero e Delisle [RD08], o GPS oferece uma excelente precisão no exterior, que é usualmente inferior a 10 metros em mais de 95% do tempo, sendo que Farid, Nordin e Ismail [FNI13] arriscam que a precisão deste sistema se encontra entre os 6 e os 10 metros, o que é bastante aceitável para o objetivo que se pretende alcançar.

2.1.2.2 A-GPS

O A-GPS diferencia-se do GPS por uma rede a ajudar nos procedimentos, permitindo assim uma poupança a nível da bateria do dispositivo e redução da quantidade de processamento que é feita no terminal.

2.1.2.3 Wi-fi

Técnicas de Wi-fi como por exemplo a *Trilateration* e *Fingerprinting* também podem ser aplicadas para determinar a posição em meios exteriores da mesma forma que em meios interiores. A combinação desta técnica com o GPS como sugeriram Li, Quader e Dempster [LQD08] produzirá resultados mais precisos. Contudo, os gastos essencialmente energéticos que esta combinação acarretará não compensarão uma vez que o GPS ou ainda melhor o A-GPS por si só já produz resultados bastante satisfatórios em meios exteriores.

Também o fato de ser uma funcionalidade presente na grande maioria dos *smartphones* facilita o acesso dos utilizadores às aplicações que dependam desta sistema.

Assim, ainda que os resultados do posicionamento por Wi-fi sejam interessantes (por exemplo por Fingerprint [LQD08]), a solução do GPS parece mais útil para o caso.

2.1.3 Google Location Services

Uma solução para implementação encontrada foram os *Google Location Services*. Este serviço, implementado pela *Google* tem como objetivo oferecer uma *framework* de alto nível que controla automaticamente fornecedores de localizações, movimento de utilizadores e precisão de localização. Também controla a calendarização de *updates* de localização baseados em parâmetros de consumo energético. Na maioria dos casos obter-se-á um melhor desempenho da bateria, assim como uma melhor precisão através deste método.⁴

A precisão da localização depende de quais das seguintes fontes de informação de localização estejam disponíveis a partir do dispositivo a partir do qual estão a ser chamados estes serviços. É possível melhorar a precisão ligando os serviços *WiFi*, especialmente quando em meios interiores onde o sinal de GPS é fraco. Não é sequer necessário estar conetado a uma rede para que a precisão melhore com o *WiFi*.

Podem ser utilizadas as seguintes fontes de informação para obter a localização do dispositivo:

- GPS: a precisão do GPS ir até vários metros dependendo do sinal de GPS e da conexão. O aparelho deve suportar GPS, tê-lo ativo permitir o seu acesso ao *Google Maps*.
- WiFi: a precisão do WiFi deve ser semelhante ao alcance de um *router* WiFi. O dispositivo deve suportar WiFi e tê-lo ligado.
- Cell ID: A precisão de Cell ID depende da densidade da "cell tower" e da informação disponível na base de dados de localização da "cell tower" da *Google*. A precisão pode ser aproximada a distâncias que vão até milhares de metros. Contudo, alguns dispositivos não suportam localização por cell ID.

Segundo este mecanismo de localização, alguns sensores, como o acelerómetro, bússola, giroscópio e barómetro, também são utilizados para melhorar a experiência de localização de uma forma geral.

É importante também referir que a área de cobertura do Cell Id e as bases de dados de localização WiFi variam por localização e não estão completas, uma vez que há zonas que não estão abrangidas por estes métodos.⁵

2.1.4 Orientação

Uma vez sabida a posição do utilizador é importante descobrir a sua orientação para saber quando e onde representar no ecrã do seu dispositivo a informação dos cupões que é pretendida.

Para tal, é necessário conjugar diversos sensores dos *smartphones* a fim de conseguir descobrir a sua orientação. Alguns desses sensores serão o de gravidade, o acelerómetro e o magnetómetro, ou então, alguns dispositivos possuem sensores sintetizados que facilitam o processo de identificação da orientação. [MS12] Com a sua utilização podemos saber de forma simples para onde está o *smartphone* do utilizador a apontar.

Em *Android* é simples obter esta informação como demonstram Varga e Kostic [VK]

⁴<http://developer.android.com/guide/topics/location/strategies.html>

⁵<https://support.google.com/gmm/answer/81873?hl=en>

2.1.5 Altitude

Quando falamos em posicionamento é preciso ter em conta um sistema de eixos segundo os quais se definirá exatamente a posição pretendida. Enquanto que em muitos casos de estudo só interessam dois desses eixos, no presente estudo interessa incluir todos os três eixos. Isto porque, a título de exemplo, em locais como centros-comerciais é importante saber o piso em que o dispositivo se encontra, para se saber que a loja que se encontra diante do dispositivo é a loja X do piso 1 e não a loja Y do piso 2 que se encontram nas mesmas coordenadas em dois dos eixos (horizontais), mas em coordenadas diferentes do terceiro eixo (vertical).

Para resolver esta questão, é possível recorrer a um dos sensores do *smartphone*, neste caso o sensor de pressão.

Sendo o GPS o sistema de posicionamento mais conhecido e utilizado atualmente, foi testada a sua precisão relativamente à altitude.

Segundo Lammel e Patzelt [GL] o GPS torna-se extremamente impreciso quando o aparelho que está a utilizar esta tecnologia não consegue detetar mais de 3 satélites ou quando existem obstáculos entre o dispositivo e os satélites, podendo esta imprecisão significar não conseguir definir com precisão suficiente o piso em que o dispositivo se encontra, colocando-o num piso diferente.

A figura 2.1 ilustra a precisão medida por um sensor GPS localizado fixo, numa sala, durante 800 segundos (cerca de 13 minutos).

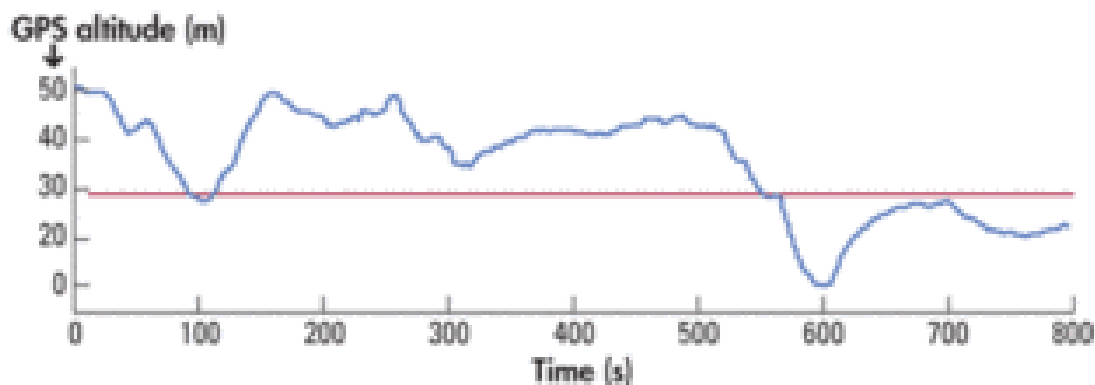


Figura 2.1: Precisão de altitude através de GPS [GL]

Como se pode observar, o erro da precisão do GPS é de cerca de 20/30m, o que é muito quando os pisos de um edifício têm por altura um valor consideravelmente menor.

Para resolver esta questão, os Esfandiyari, Mascotto e Xu [JE10] encontraram nos sensores de pressão atmosférica uma solução com uma boa relação custo/eficiência que não requer infra-estruturas adicionais para funcionar.

Configurando o sensor para um determinado valor de pressão atmosférica (neste caso 0.1mbar) permite detetar uma diferença de altitudes inferiores a 1 metro, e portanto torna-se viável distinguir pisos em edifícios com mais que um.

A figura 2.2 apresenta os dados recolhidos por um sensor de pressão localizado dentro de um edifício durante 23 minutos.

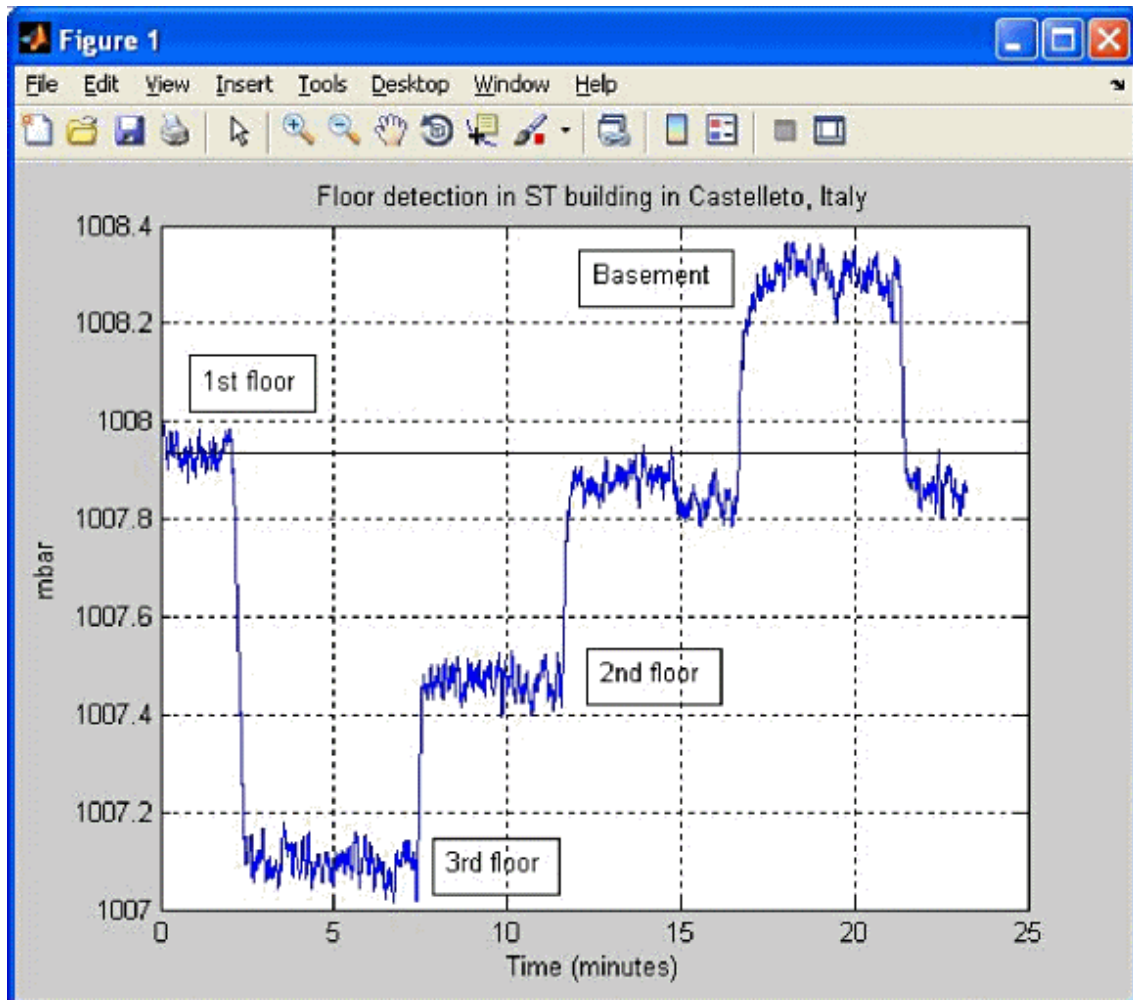


Figura 2.2: Precisão de altitude através de sensor de pressão [JE10]

Podemos verificar pela figura as diferenças da pressão atmosférica nos diversos pisos. Quanto mais alto se encontra o piso, menor é a pressão atmosférica, sendo que é visível pelo gráfico também que o erro de medição da pressão em cada piso é pequeno o suficiente para que seja possível distinguir claramente em que piso se encontra o dispositivo.

Contudo, existem algumas limitações relativamente a este sensor, bem como relativamente à sua utilização no contexto do estudo em questão.

Para o problema proposto há 2 tópicos bastante importantes. O primeiro diz-nos que o pretendido no caso de estudo é um cálculo de diferenças de altitude. Isto é, sabendo qual a altitude em que o utilizador se encontra, e a altitude onde está a "loja alvo", onde deverá ser desenhada a loja, num eixo vertical, no dispositivo?

O outro, é uma consequência do primeiro e prende-se com o facto de ser necessário guardar numa base de dados o valor da altitude de cada loja participante nas promoções e campanhas de

fidelização de que a aplicação dará informação, para se poder fazer a comparação de dados.

Para calcular a altitude a que se encontra o dispositivo é necessária a pressão atmosférica ao nível do mar. Contudo, segundo a API do Android, se esse valor não for passível de ser obtido, é possível utilizar-se o *SensorManager.PRESSURE_STANDARD_ATMOSPHERE*, que é o valor médio da pressão atmosférica ao nível do mar. Contudo, usando este valor médio perde-se precisão uma vez que o valor médio poderá ser diferente do valor real. Para além disso, estar-se-ia a ler a pressão atmosférica no dispositivo do utilizador (relativa a uma pressão atmosférica ao nível do mar) e a compará-la com a altitude da "loja alvo" obtida através da pressão atmosférica lida quando foi feita a medição (ou até mesmo de acordo com a pressão média), que podem ter valores diferentes e, portanto, dar valores de altitude bastante diferentes.

É ainda assim possível obter o valor necessário recorrendo a alguns fornecedores de previsões meteorológicas que providenciam APIs que permitem obter esta informação. Contudo, podem haver pequenas diferenças devido à distância da posição atual para onde foi medida a pressão ao nível do mar, ou devido aos intervalos em que se faz *update* da informação.

Algumas companhias permitem usar as suas APIs após ser feito um pedido por uma chave através do registo. Contudo, estas companhias limitam a utilização destes dados para um número máximo de pedidos por dia, sendo que é necessário mudar-se para um plano pago por forma a poder fazer mais pedidos. [Sam13] Desta forma, torna-se inviável a utilização deste método no presente caso de estudo. Seria contudo uma solução aparentemente bastante plausível para determinar com um bom grau de precisão a altitude a que se encontravam as lojas, permitindo assim fazer um display mais correto de cada uma das lojas em questão.

2.2 Câmara/Realidade Aumentada

A realidade aumentada oferece um novo método de interação com o mundo físico (ou real). Cria uma versão modificada da nossa realidade, enriquecida com informação digital (ou virtual), no ecrã do nosso computador ou dispositivo móvel. Juntando e combinando o virtual e o real pode alavancar um novo limite de experiência do utilizador, indo para além do que aplicações comuns são capazes. [GG13]

Uma vez tratado o posicionamento tanto do dispositivo *smartphone* como das localizações das lojas, é necessário então apresentar a informação respetiva no ecrã do aparelho no local correto.

Para este processo de realidade aumentada foram basicamente estudadas as tecnologias onde se poderia desenvolver a solução. Estas serão apresentadas a seguir.

Também foram postas algumas hipóteses de solução para o proposto envolvendo a câmara do *smartphone*. Descreve-se-ão duas possíveis abordagens a seguir.

2.2.1 Abordagens

Como referido anteriormente, foram pensadas duas abordagens para a utilização da câmara com a finalidade de descobrir os cupões de desconto das lojas. Estas técnicas envolveriam reconhecimento de imagem, que será explicado a seguir. Devido às desvantagens referidas em 2.2.1.1, entendeu-se que não traria uma vantagem significativa seguir qualquer uma destas vias uma vez que não iam de encontro a um dos objetivos *"saber os cupões específicos de cada loja para o utilizador da aplicação ainda que o utilizador não esteja diretamente em frente à loja em questão"*. Ainda assim, fala-se das duas propostas como possíveis alternativas caso se venha a verificar a possibilidade de integração de algum destes métodos na aplicação.

2.2.1.1 QR-Code



Figura 2.3: Imagem ilustrativa de um exemplo de QR-Code ⁶

A primeira prendia-se com a utilização de um código *QR-Code* junto da loja que tem cupões de desconto produzidos pela *CardMobili*. Fazendo o reconhecimento da imagem e pela identificação deste código através da câmara do dispositivo do utilizador seriam então apresentados cupões de desconto para o utilizador em questão.

- **Vantagens :** fácil de implementar.

- **Desvantagens :** pouco prático, obrigaria o utilizador a ter de estar em frente à loja para poder ter acesso aos seus cupões de desconto.

2.2.1.2 Reconhecimento Logotipo

Outra abordagem seria fazer um reconhecimento do logotipo da loja. Isto requereria ter guardados numa base de dados os vários logotipos das diferentes parceiras da *CardMobili*. Assim, o utilizador, apontando câmara do seu dispositivo para o logótipo, e tendo uma ligação à internet, teria acesso ao servidor com a base de dados, sendo que posteriormente, através de um algoritmo próprio (que seria estudado), seria feita uma pesquisa nessa mesma base de dados pelo logótipo que seria igual ao que o utilizador estaria a ver através do seu *smartphone*.

- **Vantagens :** Não envolveria custos de criação de QR-Codes e disponibilização dos mesmos nas entradas das lojas.
- **Desvantagens :** Também pouco prático, uma vez que obrigaria o utilizador a ter de estar em frente à loja para poder ter acesso aos seus cupões de desconto. Também seria algo lento, uma vez que teria de ser feita a comunicação com o servidor, corrido o algoritmo de identificação de logótipo e posteriormente a comunicação outra vez com o dispositivo enviando a informação de cupão pretendida.

2.2.2 Tecnologias

Para desenvolver a componente de realidade aumentada em Android será precisa uma ferramenta que suporte esta funcionalidade. Para o efeito, foram encontradas 2 diferentes que se parecem revelar bastante úteis. Será sobre uma destas 2 hipóteses que será desenvolvida a parte da realidade aumentada da aplicação.

2.2.2.1 AppCelerator Titanium



Figura 2.4: Logótipo do AppCelerator Titanium

O *AppCelerator Titanium* ⁷ é um ambiente extenso de desenvolvimento que serve para criar aplicações em diversos dispositivos móveis e sistemas operativos. Alguns exemplos são o iOS, Android, BlackBerry, etc.

Inclui um conjunto alargado de métodos que o tornam bastante completo e permite também a reutilização de código em novas aplicações contribuindo assim para uma poupança de tempo de desenvolvimento.

Segundo a própria companhia,

"o Titanium é a principal escolha como ambiente de desenvolvimento mobile por centenas de developers"⁷

o que permite inferir que é um ambiente de desenvolvimento fiável.

2.2.2.2 Utilização direta do Android

Existe já alguma literatura incentivando e descrevendo as técnicas necessárias para o desenvolvimento de aplicações que incluem realidade aumentada diretamente na API do sistema Android [Soo12, GG13] (e também em iOS). Uma das publicações [Soo12] mostra diversos projetos criados pelo autor nos quais enquadra imagens virtuais na imagem proveniente da câmara do dispositivo. Alguns desses projetos prendem-se com um *browser* de realidade aumentada (permite a interação do utilizador com o display exibido pela câmara, através de imagens sobrepostas neste *display*). Algumas das técnicas apontadas podem ser utilizadas em novos projetos, diretamente no sistema Android, com garantia de boa performance. Esta facilidade e a possibilidade de maior performance levaram-nos a escolher o desenvolvimento direto em Android para a criação de uma aplicação que será a prova de conceito da solução proposta.

⁷<http://www.appcelerator.com/titanium/>

2.3 Trabalhos Relacionados

Existem várias aplicações que têm funcionalidades que interessavam para o tema. Falamos aqui brevemente de duas delas.

2.3.1 Layar



Figura 2.5: Imagem do Logótipo do Layar ⁸

O *Layar*⁹ é um *mobile* browser que permite aos utilizadores encontrar diversos itens perto destes baseado em realidade aumentada através da câmara do *smartphone*.

A ideia do Layar passar por usar uma combinação da câmara do dispositivo com a bússula e GPS para identificar a posição do utilizador e campo de visão, devolver informação baseada nessas coordenadas geográficas, e sobrepor essa informação na vista da câmara.

É possível, através desta aplicação, utilizar objetos 3D para os colocar em sítios específicos desde que a câmara esteja a apontar para esse local. Ou também, tocar um determinado elemento de audio designado para tocar quando a câmara estiver a apontar para um sítio em específico.

Comparativamente à aplicação que se pretende desenvolver na dissertação o *layar* funciona essencialmente sobre a área da realidade aumentada e posicionamento *outdoor*, tendo uma lacuna em meios *indoor* uma vez que o seu sistema de funcionamento funciona apenas por GPS, que como vimos anteriormente neste artigo tem uma precisão baixa nestas situações.

2.3.2 Junaio + Valpak



Figura 2.6: Imagem dos Logótipos da Junaio e Valpak ¹⁰¹¹

⁹<http://www.infinity-layar.info/what-is-layar.html>

Junaio é um *browser mobile* de realidade aumentada, enquanto que a *Valpak* [Kat11] é uma empresa que trabalha com cupões de desconto, à semelhança da empresa proponente deste tema de dissertação - *CardMobili*.

Depois de os utilizadores transferirem a aplicação da *Junaio*, poderão ver cupões, poupanças, e propostas de negócios e rebater os que lhes interessam mostrando ao vendedor da loja o cupão que têm na aplicação.

Um exemplo de como funciona a aplicação de realidade aumentada é apresentado na figura 2.7. In addition to the application, Valpak also offers deals via an application as well as offers



Figura 2.7: Imagem ilustrativa da interface da aplicação da *Junaio*

SMS deals. Para além do descrito, a *Valpak* também oferece propostas através da aplicação, bem como propostas por SMS.

Comparativamente ao caso de estudo apresentado neste relatório, a *Valpak+Junaio* fizeram uma aplicação muito semelhante, contudo não fornece informação para se saber como funciona em ambientes *indoor*, podendo ser essa a grande diferença para a aplicação proposta aqui apresentada.

2.4 Conclusões

No presente capítulo foi feita uma revisão do estado de arte para o tema do relatório. Começou-se por abordar as questões de posicionamento em duas dimensões. Resumindo os dados que foram adquiridos, foi obtida a seguinte tabela [FNI13] 2.8:

Como podemos observar, para meios interiores, apenas os algoritmos de *fingerprint* e TDoA apresentam uma elevada precisão, enquanto que os restantes ficam um pouco mais atrás. Dentro destes 2, podemos reparar que TDoA é afetado por caminhos-múltiplos, enquanto que *fingerprint* não. O custo deste último algoritmo também é mais acessível do que a grande maioria dos algoritmos estudados.

Desta forma, podemos corroborar a informação de Jian [Jia12] quando afirma que WLAN RSS Fingerprinting tem-se tornado uma das técnicas mais exploradas em localização *indoor* por ser um método relativamente fácil de pôr em execução e que é tolerante ao ruído de sinal e, portanto, consegue assim atingir resultados mais precisos.

Estado da Arte

TABLE 1: Comparisons of indoor position methods.

Method	Measurement type	Indoor accuracy	Coverage	Line of sight (LOS)/nonline-of sight (NLOS)	Affected by multipath	Cost	Notes
Proximity	Signal type	Low to high	Good	Both	No	Low	(1) Accuracy can be improved by using additional antenna. However, it will increase the cost. (2) Accuracy is on the order of the size of the cells.
Direction (AoA)	Angle of arrival	Medium	Good (Multipath issues)	LOS	Yes	High	(1) Accuracy depends on the antenna's angular properties. (2) Location of antenna must be specified.
Time (ToA, TDoA)	Time difference of arrival	High	Good (Multipath issues)	LOS	Yes	High	(1) Time synchronization needs. (2) Location of antenna must be specified.
Fingerprinting	Received signal strength	High	Good	Both	No	Medium	(1) Need heavy calibration. (2) Location of antenna is not necessary.
Dead reckoning	Acceleration, velocity	Low to medium	Good	NLOS	Yes	Low	Inaccuracy of the process is cumulative, so the deviation in the position fix grows with time.
Map matching	An algorithm based on algorithms based on projection and pattern recognition	Medium	Medium (indoor) Good (outdoor)	NLOS	Yes	Medium	(1) Map matching purely focus on algorithms and not fully on position methods, coordinate transformation, and geocoding. (2) Using pattern recognition, high computing complex and poor real time issue occur.

Figura 2.8: Imagem representativa de uma tabela com os diversos algoritmos indoor estudados [FNI13]

No entanto, podemos também concluir que o método de *fingerprint* precisa de cuidados de calibração que podem tornar a sua manutenção um processo custoso.

Ainda relativamente às técnicas de posicionamento, também foram estudadas as alternativas de GPS, RFID e Wi-fi. Na seguinte tabela [FNI13] 2.9:

Podemos observar que a precisão de RFID é bastante boa e seria até ideal. Contudo, Chon, Jun, Jung e An [CJJA04] afirmam que

"Esta técnica, não substituirá o GPS. Em vez disso será uma técnica complementar."

Além disso implicaria a instalação das tags RFID em vários locais e lojas aderentes.

Utilizar as 2 tecnologias torna-se então desnecessário uma vez que o GPS, sendo bastante acessível a qualquer utilizador, também consegue obter resultados bastante satisfatórios.

Relativamente ao Wi-fi, seria uma melhor solução, contudo está dependente da existência de sinais para poder fazer os seus cálculos. Portanto, a área de cobertura do Wi-fi é bastante inferior à do GPS, não compensando portanto a utilização isolada desta técnica.

Também foram estudados os *Utilização dos Google Location Services* que segundo a investigação feita permite obter a localização pretendida tanto em meios interiores como exteriores, através da conjugação das técnicas de GPS e de Wi-Fi, permitindo assim um elevado grau de precisão num panorama mais geral de localizações possíveis a nível global. Permite também fazer uma gestão de energia algo personalizável pelo utilizador tornando-se assim este método um forte candidato a ser implementado.

Estado da Arte

TABLE 3: Comparison of common position systems used for localization.

System	Accuracy	Principles used for localization	Coverage	Power consumption	Cost	Remarks
GPS	6 m–10 m	ToA	Good outdoor Poor indoor	Very high	High	(1) Satellite based Positioning. (2) Processing time and computation is slow.
WiFi	1 m–5 m	Proximity, ToA, TDoA, RSSI Fingerprinting, and RSSI theoretical propagation model	Building level (outdoor/indoor)	High	Low	(1) Infrastructure available everywhere. (2) Initial deployment is expensive. (3) Multipath susceptible slightly.
RFID	1–2 m	Proximity, TOA, RSSI theoretical propagation model	Indoor	Low	Low	(1) Real time location system. (2) Response time is high. (3) Manual programming.

Figura 2.9: Imagem representativa de uma tabela com os diversos algoritmos outdoor estudados [FNI13]

De seguida abordou-se as questões de orientação e altitude e concluiu-se que os sensores que hoje em dia já existem nos *smartphones* conseguem produzir resultados bastante satisfatórios para o efeito.

Tratou-se também a questão da realidade aumentada e da câmara. Falou-se de duas possíveis abordagens de utilização da câmara do dispositivo e/ou realidade aumentada. Serão duas soluções que ficam em aberto caso venham a ser interessantes de voltar a abordar, contudo, para já, concluiu-se que a utilidade destas abordagens poderia ser contornada com as técnicas de posicionamento.

Por fim, reviram-se algumas tecnologias que ajudarão no desenvolvimento da aplicação de realidade aumentada, e algumas tecnologias que estão de alguma forma ligadas com o tema.

Capítulo 3

Arquitetura

A empresa *Cardmobili* dispõe já de uma aplicação denominada precisamente *Cardmobili* cujo objetivo principal é servir de carteira digital, levando os utilizadores a ter cada vez menos cartões físicos nas suas carteiras e mais cartões digitais nesta aplicação. Para além disso, esta aplicação tem propostas de ofertas para os utilizadores, como cupões de descontos em determinadas lojas ou promoções direcionadas para si através do seu cartão de fidelização com uma loja. Pretendeu-se então desenvolver um novo módulo para a aplicação, integrando, como indicações para o utilizador, gráficos inseridos na imagem real da câmara, informando acerca da proximidade e posicionamento de lojas para as quais se dispõe de cupões ou outros elementos de fidelização. Apresentam-se então os principais requisitos e funcionalidades que este novo módulo deve ter, satisfazendo as necessidades do utilizador. É depois descrita a arquitetura base do módulo para se conseguir fazer uma implementação simples, capaz de ser integrada em qualquer aplicação Android, e que atinja os resultados pretendidos. A arquitetura abordada neste capítulo refere apenas a parte que diz respeito a este módulo, que utiliza técnicas de realidade aumentada, abordando qual a sua interação com a aplicação principal mas sem detalhar como foi desenvolvida a arquitetura desta.

3.1 Requisitos e Funcionalidades

Para conseguir dar resposta às necessidades dos utilizadores, é importante ter em conta alguns objetivos e funcionalidades mais concretos relativamente ao que se pretende desenvolver, mais aprofundados do que os que foram mencionados na introdução. Alguns destes serão de elevada importância e outros serão considerados pontos extra que se poderia desenvolver para complementar a aplicação.

Foram identificados os seguintes objetivos como sendo principais no desenvolvimento de um protótipo:

- Identificar com precisão a localização dos utilizadores e lojas em meios interiores e exteriores.
- Mostrar corretamente, no ecrã, a localização da loja num eixo horizontal, e uma aproximação da localização da loja segundo um eixo vertical.

- Exibir de forma agradável para o utilizador, a identificação da loja e informação relevante, como o nome da loja, logótipo do comerciante, distância a que se encontra da loja e número de pontos que o utilizador tem no cartão para aquela loja.
- Integrar o módulo que se vai desenvolver com a base de dados já existente da *Cardmobili*, por forma a obter as localizações reais das lojas.
- No caso de haver sobreposição de lojas no ecrã, exibir uma identificação de agrupamento de lojas, possivelmente adicionando um ícone que signifique explicitamente que existe mais de uma loja naquele lugar, para que o utilizador saiba que tem de seleccionar esse ícone para expandir as lojas "escondidas".
- Prever a situação em que o utilizador está sem internet e portanto não consegue pedir ao servidor a localização das lojas.

Para além destes objetivos principais, foram definidos alguns pontos extra. Dado o limite de tempo para a realização do trabalho de dissertação, entende-se que algumas funcionalidades poderão ser acrescentadas ao protótipo depois de este estar funcional. Estas são:

- Dar a possibilidade ao utilizador para mudar entre uma vista de Realidade Aumentada para uma vista de lista (onde os cupões serão exibidos, ordenados por proximidade, numa lista).
- Dar a possibilidade ao utilizador para mudar entre uma vista de Realidade Aumentada para uma vista de mapa (onde o utilizador poderá ver num mapa os locais onde os cupões poderão ser redimidos).
- Alternativamente ao ponto anterior, também poderá ser possível mudar entre vistas de acordo com a inclinação/rotação do dispositivo.
- Implementar, no módulo desenvolvido, a medição correta da altitude tanto do utilizador como das lojas. Isto permitirá ao utilizador ter a melhor localização possível duma loja mesmo em pisos diferentes de um edifício.
- Um radar que mostraria os cupões mais próximos relativamente ao utilizador. Isto permitira ao utilizador ter uma melhor perspectiva de onde se localizam os pontos de interesse perto de si.

3.2 Perspetiva Geral

Na figura 3.1 mostra-se um esquema simples de como o módulo desenvolvido se integra com o restante da aplicação (*Cardmobili*). Este módulo deverá ter acesso e interagir com o restante da aplicação para navegação e consulta da sua base de dados.



Figura 3.1: Esquema base de interação entre as diversas componentes da aplicação.

3.3 Servidor e Aplicação

A comunicação entre a aplicação que vai albergar o módulo de realidade aumentada e o servidor já está feita pela Cardmobili. Contudo, para o desenvolvimento do módulo, será necessário fazer diversos pedidos ao servidor como por exemplo das lojas que serão para mostrar no dispositivo.

Para conseguir obter a informação pretendida que depois será utilizada no módulo, foi necessário implementar no servidor APIs REST que servirão para fazer a comunicação pretendida entre o módulo e o servidor. Desta forma, tornou-se possível obter a informação necessária sobre as várias lojas através do módulo.

3.4 Módulo de Realidade Aumentada

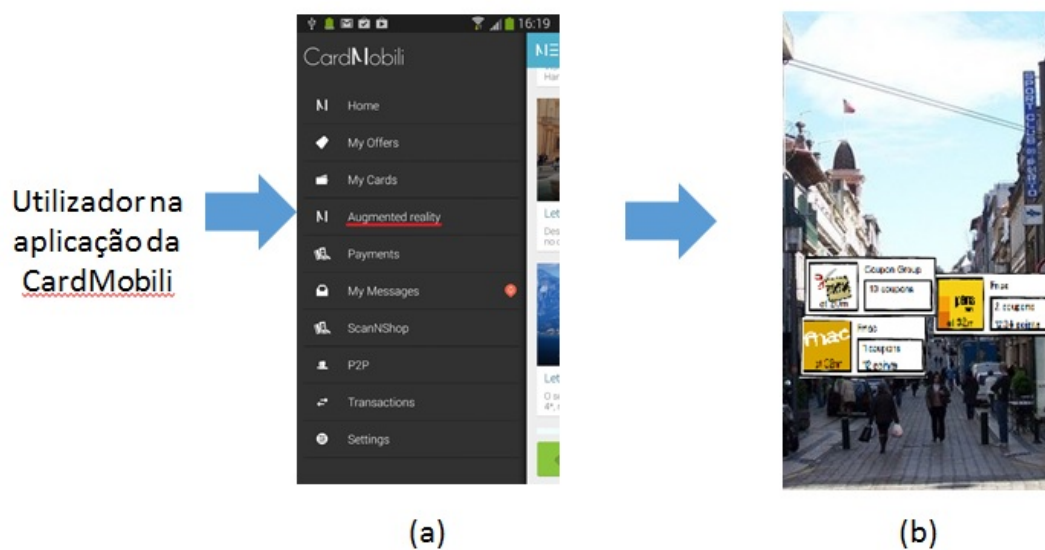
O módulo implementado necessita comunicar com a aplicação a fim de saber que cupões de desconto deve exibir. De igual modo, para comunicar com o servidor, são executadas chamadas através da aplicação *Cardmobili*, nomeadamente o módulo de realidade aumentada faz um pedido à aplicação mãe para obter as lojas que existem nas imediações (estas são definidas pelo utilizador quando define o raio sobre o qual deve ser feita a pesquisa) e esta por sua vez fará o mesmo pedido ao servidor. A decisão de construir o código desta forma, a nível de arquitetura, resulta

do facto de a aplicação já existente dispor já de camada de *webservices*, ficando assim toda a lógica centralizada. Este diálogo entre o módulo de realidade aumentada e da aplicação é feito por intermédio de uma interface, como será explicado no capítulo de implementação.

É também no protótipo implementado que serão feitos todos os cálculos e ajustes necessários para o mostrar uma representação gráfica das lojas no dispositivo, como por exemplo o cálculo da posição horizontal e vertical, no ecrã, onde deve ser exibida a loja.

3.5 Fluxo do Módulo

Na figura seguinte representa-se o fluxo de *interfaces* com o utilizador desde que se entra na aplicação *Cardmobili* até ao módulo de realidade aumentada que se aborda nesta dissertação.



Arquitetura



Figura 3.2: Imagem ilustrativa da interface esperada da aplicação

O utilizador abre a aplicação da *Cardmobili* e no painel lateral esquerdo terá os diferentes módulos que pode utilizar sendo um deles, o de realidade aumentada (figura 3.2a).

Abrindo o módulo de realidade aumentada, é imediatamente exibida a imagem da câmara do dispositivo. Quando o utilizador apontar para a mesma direção de uma loja existente na base de dados, deve aparecer no ecrã uma identificação da loja em questão, com alguma informação básica sobre a mesma (figura 3.2b). Se mais que uma loja estiver na direção da câmara, então também deverão ser exibidas.

Clicando com o dedo na imagem de uma loja que estiver a ser exibida no ecrã, pode acontecer uma de 3 situações:

- Se o utilizador possui um cartão de fidelização para a loja que está a ser exibida no ecrã, então deverá ser aberta a interface representada na figura 3.2c que representa a informação dos descontos na loja, para o respetivo cliente, com base no seu cartão de fidelização.
- Se o utilizador carregar numa identificação, no ecrã, que represente um conjunto de cupões (quando há vários cupões sobrepostos, deve ser exibida uma identificação através de um ícone representativo de um agrupamento de cupões), então deve ser aberta a interface demonstrada na figura 3.2d. Esta é uma lista com os vários cupões de desconto presentes naquele agrupamento.
- Há ainda a hipótese de a identificação no monitor se referir apenas a um único cupão de desconto para o utilizador. Nesse caso, deve ser aberta imediatamente a informação relativa

ao cupão em questão, como ilustrado na figura 3.2e.

De notar que as interfaces ilustradas na figura 3.2c), d) e e) fazem parte da aplicação *Cardmobili*, estando já implementadas. Compete ao módulo executar a comunicação entre as duas componentes a fim de obter a resposta pretendida e exibir a informação pretendida.

Contudo, a aplicação para funcionar deverá necessitar de uma ligação de dados WiFi ou por Dados Móveis pelo menos quando o programa for iniciado, para que seja feita a comunicação com o servidor e se obtenha informação das lojas mais próximas e dos cupões. Assim, a primeira validação que deve ocorrer, verificará se o utilizador tem as comunicações *WiFi* ou por Dados Móveis ativadas. Se não tiver, a aplicação não consegue comunicar com o servidor para receber a informação inicial sobre as lojas e cupões, devendo, nesse caso, ser apresentada uma notificação ao utilizador para que este saiba que tem de ativar pelo menos uma das duas opções (figura 3.3).

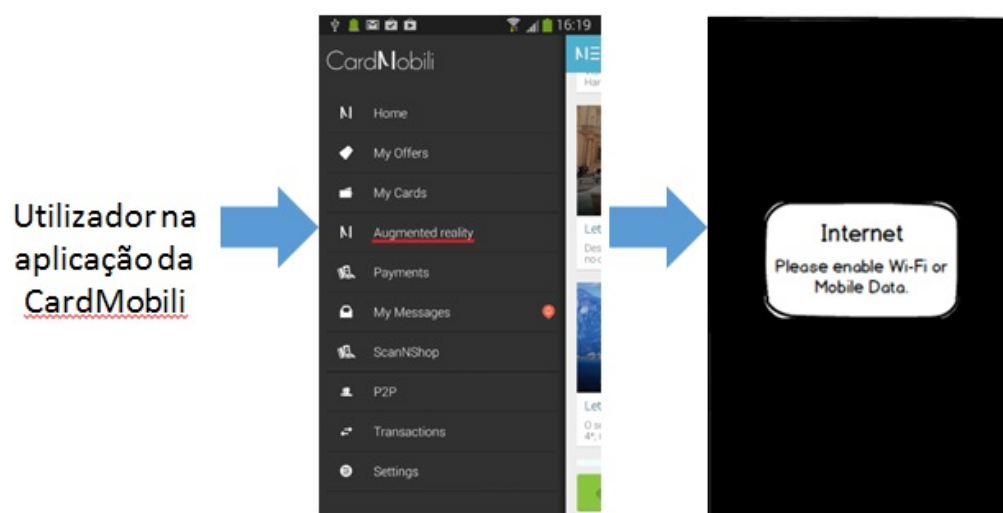


Figura 3.3: Imagem ilustrativa da interface esperada da aplicação

3.6 Sequência de Operações

No diagrama de sequência seguinte mostra-se a sequência típica de operações para um melhor entendimento de como funciona a aplicação a nível interno.

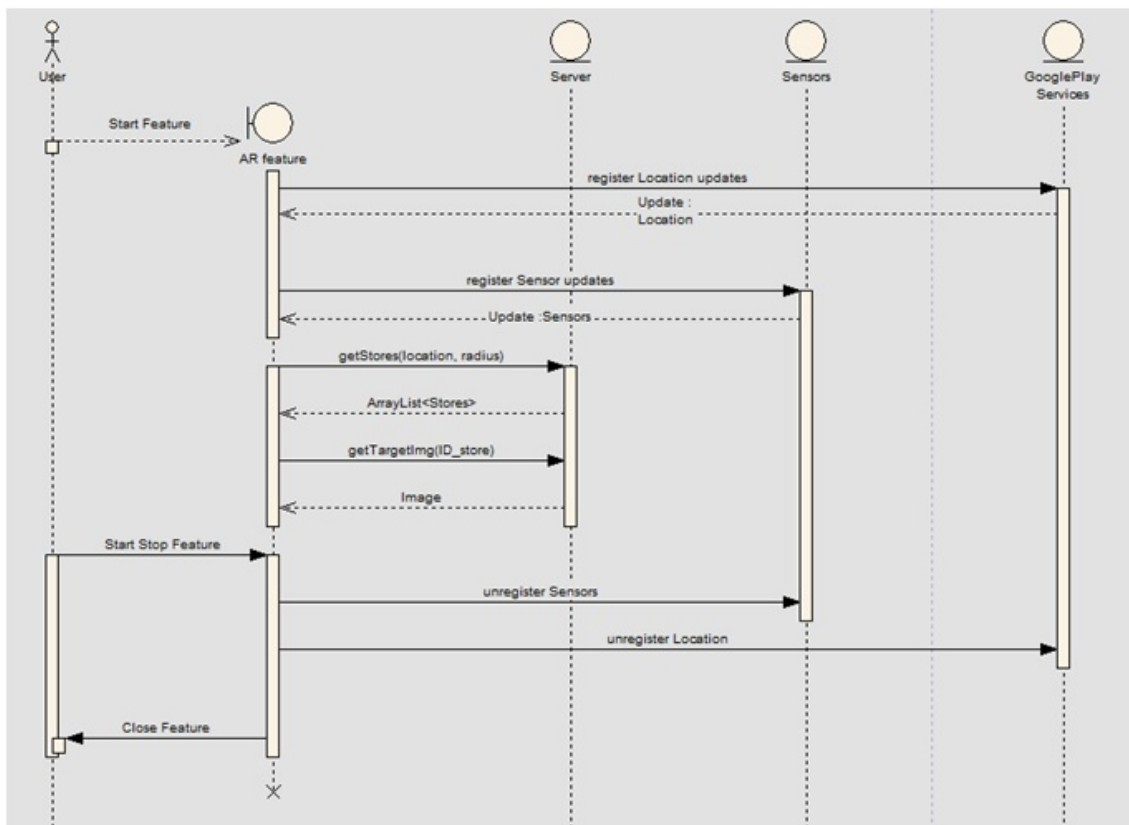


Figura 3.4: Imagem ilustrativa do diagrama de sequência esperado para o módulo

O utilizador, quando inicia o módulo de realidade aumentada vê imediatamente a vista da câmara onde a realidade aumentada deverá depois ocorrer. Simultaneamente, devem ser executados os seguintes procedimentos:

- O sistema de localização e os sensores são iniciados e irão continuamente atualizar a informação sobre a orientação, inclinação e posicionamento do dispositivo sempre que for detetado um deslocamento considerável por parte do utilizador.
- Depois, é necessário verificar que lojas que existem nas imediações num raio que é especificado em `getStores(location, radius)`.
- Uma vez obtidas as lojas circundantes, é possível perguntar ao servidor que cupões destas lojas existem para que se mostrem na visualização do ecrã, de modo a poupar recursos ao servidor.
- Ao mesmo tempo, e uma vez que as lojas podem ter um logótipo associado, é também pedido ao servidor a imagem desse logótipo das lojas que estão a ser exibidas no ecrã.
- Por fim, quando o utilizador sinaliza que pretende fechar a aplicação, fecha-se o registo das atualizações de localização e dos sensores e libertam-se esses recursos para se fechar de vez a aplicação.

Capítulo 4

Implementação

No presente capítulo serão indicadas e explicadas as decisões que foram tomadas no decorrer do trabalho, bem como serão fornecidos mais alguns detalhes de pontos previamente mencionados na arquitetura da solução.

É importante recordar que um dos objetivos é a implementação de um módulo versátil que possa ser integrado noutras aplicações *Android* se assim for pretendido.

4.1 Tecnologias

De acordo com o estudo feito no estado de arte, o ambiente de desenvolvimento *AppCelerator Titanium* aparentou ser extremamente útil e adequado para a resolução do problema em questão e por isso, houve uma tentativa de iniciar a implementação recorrendo a este ambiente. Contudo, devido à modularidade que se pretendia dar, tornando o módulo implementável a partir de outra aplicação já existente nomeadamente em *Android* (por requisito da empresa), descobriu-se não ser possível dar continuidade ao desenvolvimento usando esta plataforma. Tendo em conta informação fornecida por peritos na área¹, concluiu-se que não era possível criar uma aplicação neste ambiente que pudesse ser integrada numa aplicação *Android*. Desta forma, a *Cardmobili* decidiu abandonar o desenvolvimento nesta plataforma e trabalhar diretamente no *Android* nativo através da ferramenta de desenvolvimento *Eclipse*.

4.2 Realização

Seguindo o esquema geral da arquitetura apresentado na figura 3.1, aborda-se nesta secção as opções tomadas relativamente a uma das partes principais bem como alguns pontos da implementação feita para atingir os objetivos propostos.

¹<http://developer.appcelerator.com/question/162818/merging-existing-android-app-with-new-titanium-feature>

4.2.1 Módulo de Realidade Aumentada

4.2.1.1 Estrutura Geral

Uma vez que o objetivo principal passa pela sobreposição de imagens virtuais sobre uma imagem real (captada pela câmara do dispositivo), a implementação foi dividida em dois blocos principais:

- *AugmentedMain* - onde foi feita a comunicação entre os diversos módulos (aplicação mãe e servidor), e inicializados os sensores necessários para o desenvolvimento do protótipo e a câmara.

Esta classe estende a classe *Fragment* do *Android*. Esta opção foi tomada por forma a permitir a integração do módulo de realidade aumentada numa das atividades da aplicação mãe. Assim, basta a aplicação principal incluir o fragmento do protótipo para poder ter as funcionalidades neste desenvolvidas.

- *AugmentedView* - onde são feitos os cálculos de posicionamento necessários para fazer a exibição no local certo do ecrã do dispositivo e onde é feito o *overlay* da informação pretendida sobre a imagem da câmara.

Esta é uma classe que estende a classe *View* que faz já parte do *Android*. Esta é uma classe que permite a construção de uma *interface* por parte do programador bem como a interação com ela através de interação, como por exemplo o evento de toque no ecrã do dispositivo.

Para além destas duas classes, também é importante referir uma terceira que será o elo de comunicação entre as estruturas da aplicação *Cardmobili* com o módulo de realidade aumentada. Essa classe foi denominada *AugmentedRealityInterface*, e será explicada a sua utilidade mais à frente.

4.2.1.2 Localização

De acordo com o estudo feito no capítulo 2, assumiu-se ser mais prático, fiável e célere a implementação do sistema de localização baseado no *Google Location Services*. A versatilidade apresentada por este serviço foi um fator importante na escolha, uma vez que permite ao utilizador optar por diferentes graus de precisão e gestão de bateria, baseando-se apenas no *WiFi*, ou no GPS ou na junção dos 2, aumentando os consumos de bateria em cada caso respetivamente. Outro fator importante foi a facilidade de implementação deste processo comparativamente aos restantes devido às *APIs* simples da *Google*.

De acordo com Tsuwei Chen², o nível de precisão dos processos e algoritmos recentemente implementados nos *Google Location Services* é maior do que os implementados em versões mais antigas como se pode ver na figura 4.1. Esta informação foi também importante para a decisão tomada relativamente ao uso desta metodologia para o cálculo da localização do utilizador na implementação do módulo previsto.

²http://static.googleusercontent.com/media/www.google.com/pt-PT/intl/zh-CN/events/facultysummit/2010/files/mobile_location.pdf

Implementação

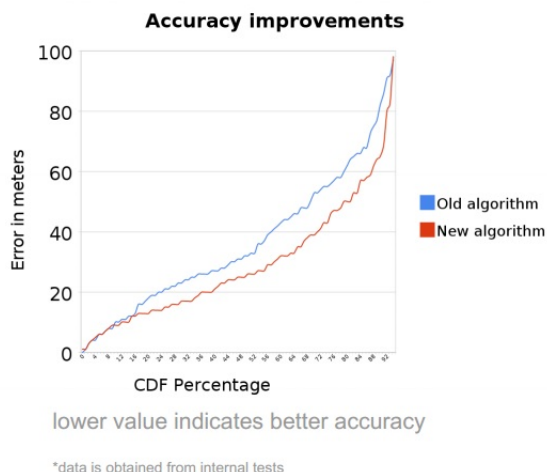


Figura 4.1: Comparação entre os antigos processos de cálculo da localização e os mais recentes existentes na última versão do *Google Location Services*²

4.2.1.3 Sensores

Para além da localização do utilizador, é também importante saber para onde está a câmara a apontar e qual a sua inclinação de acordo com os diversos eixos definidos no dispositivo como representado na figura 4.2. Assim, de acordo com o estudo feito, é necessário utilizar os sensores de orientação disponíveis no dispositivo para determinar estes dados.

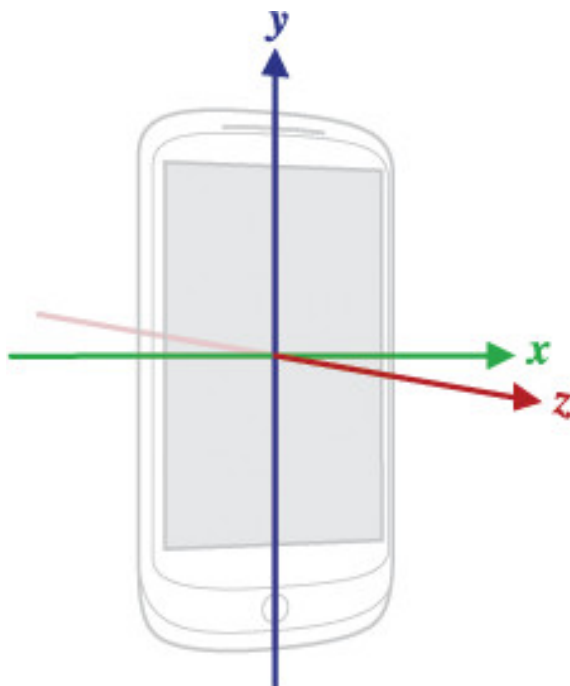


Figura 4.2: Representação dos 3 eixos num dispositivo móvel [Dab11]

Implementação

Através do *Android* nativo é fácil aceder aos valores fornecidos pelos sensores e com frequências relativamente elevadas (na ordem de alguns Hz). Foi então utilizado o sensor de orientação do dispositivo. Este sensor utiliza internamente o giroscópio, o sensor de campo magnético e por vezes o acelerómetro e está geralmente implementado em muitos dispositivos. Através deste sensor conseguimos obter três medidas[Dab11]:

- *azimuth* - é medido em graus, de 0° a 360° . Descreve a rotação em torno do eixo dos Z, em que 0° representa o dispositivo ter o eixo Y a apontar na direção do Norte geográfico. Assim, se este valor for 0° , o topo do dispositivo está a apontar para Norte, se for 90° , está a apontar para Este. A mesma direção de rotação se aplica para os restantes valores que este campo pode assumir.



Figura 4.3: Ilustração representativa do eixo *azimuth*[Dab11]

- *pitch* - é medido em graus. Esta indicação representa uma rotação em torno do eixo dos X, ou seja, o ângulo entre o eixo Y do dispositivo e a posição horizontal. Este ângulo pode assumir valores entre -180 e $+180$.



Figura 4.4: Ilustração representativa do eixo *pitch*[Dab11]

- *roll* - também é medido em graus. Mostra a rotação em torno do eixo dos Y, ou seja, a

Implementação

relação entre a posição horizontal do chão e o eixo X. Pode assumir valores que variam desde -90 até 90.

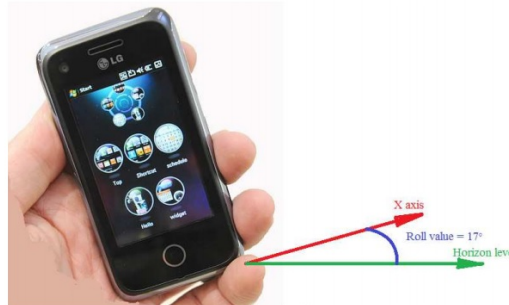


Figura 4.5: Ilustração representativa do eixo *roll*[Dab11]

Uma vez extraída esta informação, mais a informação da localização do utilizador, sabem-se quase todos os dados necessários para se poder descobrir se uma loja está no ângulo de visão captada pela câmara ou não. Mais à frente falar-se-á da informação que falta para conseguir o pretendido.

4.2.1.4 Informação das lojas

Na reprodução de imagens das lojas sobre a imagem da câmara dever-se-á incluir informação específica sobre as lojas como o nome, número de pontos ou o número de cartões que o utilizador tem daquela loja. Para além desta informação, também é preciso saber para cada loja onde é que vai ser desenhada esta informação no ecrã. Ao mesmo tempo, é preciso estar constantemente a atualizar a informação dos dados dos sensores e de quais as lojas que rodeiam o utilizador (sempre que houver uma atualização da localização).

Face ao exposto, tomou-se a decisão de que estruturalmente se teriam 2 objetos distintos para representar as lojas.

- O *StoreData*, que estende o objeto *Store* apresentado na secção 3.3 do capítulo 3.
- O *StoreDisplay*, criado com o intuito de guardar a informação necessária para fazer mostrar a informação da loja no ecrã.

Desta forma são evitados problemas de concorrência provocados por acessos simultâneos ao mesmo objeto quando este está a ser atualizado e ao mesmo tempo a ser acedido para se obter a informação necessária para fazer o *display* desta.

4.2.1.5 Posicionamento no ecrã

Com toda a informação calculada até agora, falta apenas saber qual o campo de visão da câmara, em graus, para saber onde, no ecrã, se deve exibir as informações das lojas que se encontram na direção para onde a câmara está a apontar.

Este campo de visão é facilmente obtido no *Android* através dos parâmetros do objeto *Camera*.

Implementação

```
1 protected void updateCameraAngles(android.hardware.Camera camera) {  
2     horizontalAngle = camera.getParameters().getHorizontalViewAngle() / 2;  
3     verticalAngle = camera.getParameters().getVerticalViewAngle() / 2;  
4 }
```

Neste caso, os valores foram divididos por 2 para obter o valor do campo de visão à esquerda e à direita do eixo ótico representativo do ponto para onde o dispositivo está a apontar.

Uma vez obtido este valor, foi então possível calcular a posição horizontal onde a informação das lojas deve ser exibida. O *array* de lojas é percorrido e, loja a loja, é verificado se a mesma deve ou não ser exibida no ecrã. De notar que o *array* de lojas contém apenas as lojas retornadas pelo servidor num raio de distância especificado pelo utilizador.

Dado que o *azimuth* tem um valor entre 0 e 360°, o posicionamento do dispositivo para 350° implica captar todas as imagens que se encontrem entre 350-horizontalAngle e 350+horizontalAngle. Contudo, 350+horizontalAngle será maior que 360 quando na verdade deveria reiniciar no valor 0.

Para resolver esta situação, calculam-se os limites do lado esquerdo e direito da câmara e convertem-se para o valor correto caso seja necessário.

```
1 if (leftLimit < 0)  
2     leftLimit += 360;  
3  
4 if (rightLimit > 360)  
5     rightLimit -= 360;
```

Posto isto, é necessário prever 2 situações:

- 1 - a soma ou subtração do horizontalAngle com o *azimuth* ultrapassa o valor 360 ou passa a valores negativos, respetivamente.
- 2 - a soma ou subtração do horizontalAngle com o *azimuth* não tem influência no resultado e o cálculo pode ser feito diretamente.

Implementação

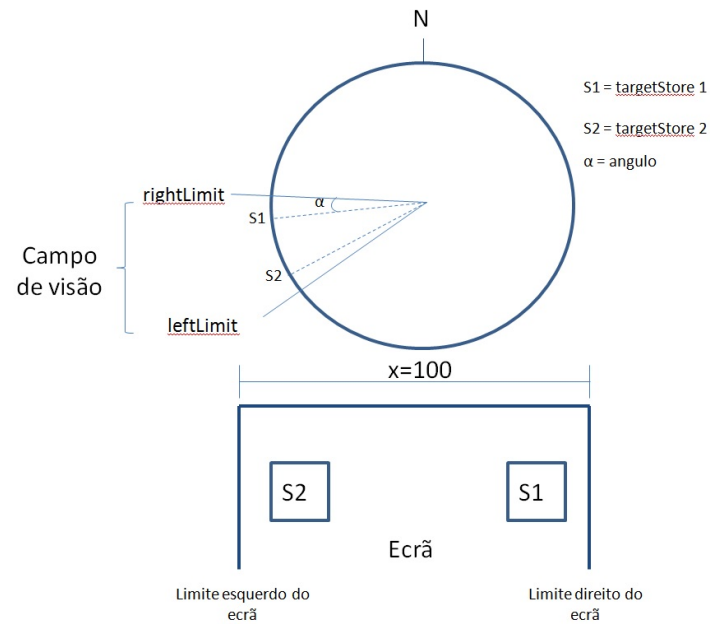


Figura 4.6: 1) Situação em que o campo de visão ultrapassa o valor 360 ou tem valores negativos

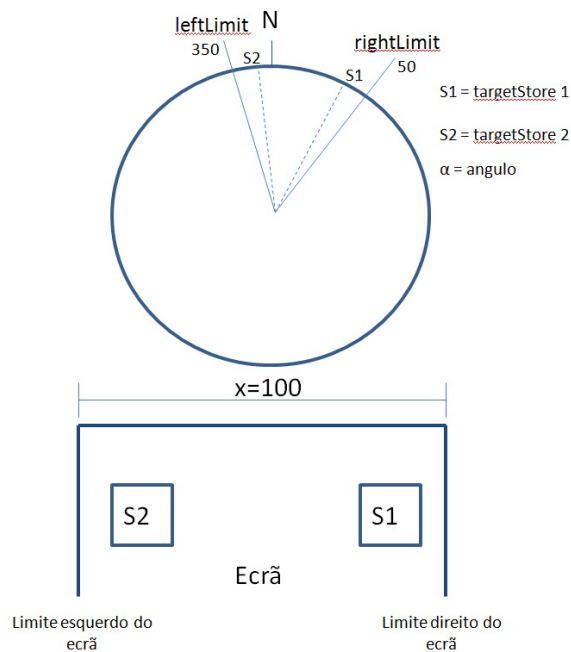


Figura 4.7: 2) Situação em que o campo de visão não ultrapassa o valor 360 ou tem valores negativos

Implementação

Para saber qual das 2 está a ocorrer, basta testar se o limite do lado direito tem um valor superior ao do lado esquerdo:

```
1 if (leftLimit < rightLimit) {  
2   ...  
3 }
```

Neste caso (situação 2), é necessário verificar se a loja (*targetStore*) está dentro do campo de visão do dispositivo. Caso esteja, é preciso descobrir onde deve ser exibida no ecrã. Para isso, calcula-se primeiro o ângulo formado entre o limite direito de visibilidade da câmara e a loja. A partir daqui, descobre-se a percentagem do campo de visão correspondente a esse ângulo, sabendo que 100% se refere ao ângulo total do campo de visão da câmara do dispositivo.

```
1 angulo = rightLimit - targetStore;  
2 percent = angulo * 100 / (horizontalAngle * 2);  
3 x = (100 - percent) / 100;
```

A posição *x* onde a informação deve ser exibida é também uma percentagem onde o valor 100 em "(100-percent)" representa 100% do ecrã, ou seja, representa o limite direito do ecrã.

Uma vez que se calculou o ângulo formado entre o limite direito do campo de visão e a localização da loja, então a posição *x* deve subtrair ao lado direito do ecrã o valor percentual representativo de onde se encontra a loja (100-percent). A figura ?? representa o caso descrito.

O resultado da implementação fica então o seguinte:

```
1 if (leftLimit < rightLimit) {  
2   if (targetStore > leftLimit && targetStore < rightLimit) {  
3     angulo = 0;  
4     percent = 0;  
5     // calculate where the store will be displayed on screen  
6     angulo = rightLimit - targetStore;  
7     percent = angulo * 100 / (horizontalAngle * 2);  
8     x = (100 - percent) / 100;  
9     horizontalDraw = true;  
10  } else  
11    horizontalDraw = false;  
12 }
```

No caso da situação 1, em que a soma ou subtração do *horizontalAngle* com o *azimuth* ultrapassa o valor 360 ou passa a valores negativos, é preciso fazer alguns cálculos diferentes.

É sabido que o valor do ângulo entre a loja e o Norte geográfico será sempre um valor entre 0 e 360°. Assim, para a situação prevista, sabe-se que a loja se encontra no campo de visão do dispositivo se o valor do ângulo da loja for superior ao limite esquerdo deste campo, ou inferior

Implementação

ao limite direito.

Caso se verifique, há agora 2 situações que se devem prever:

- a - A situação em que o ângulo da loja é maior que 0 mas menor que o limite direito do campo de visão.
- b - A situação em que o ângulo da loja é maior que 0 mas o limite do campo de visão é menor do que o ângulo da loja.

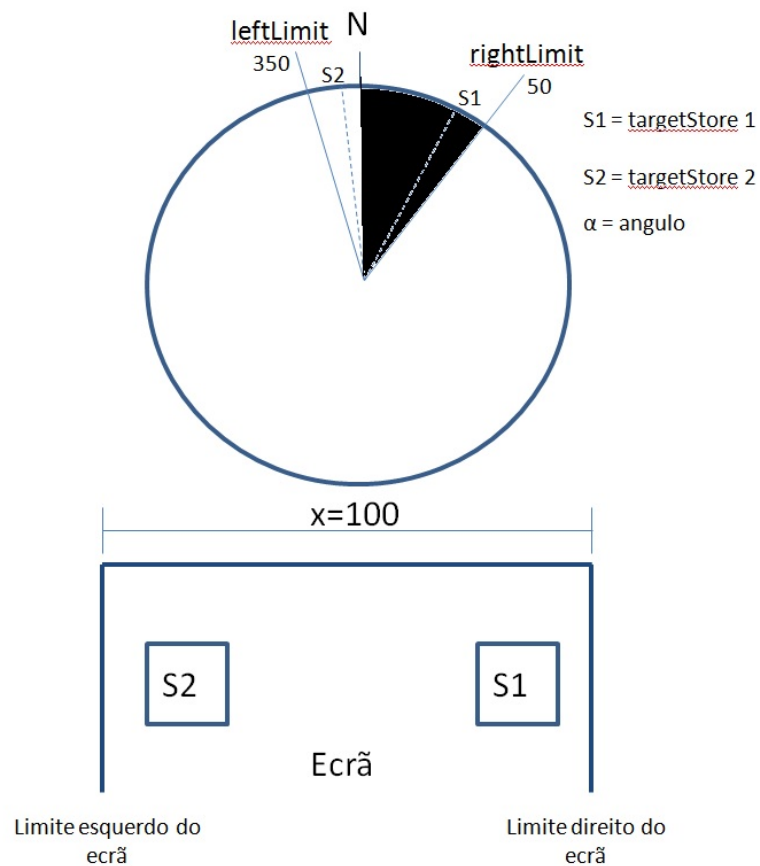


Figura 4.8: a) Ângulo da loja maior que 0 e menor que limite direito do campo de visão

Implementação

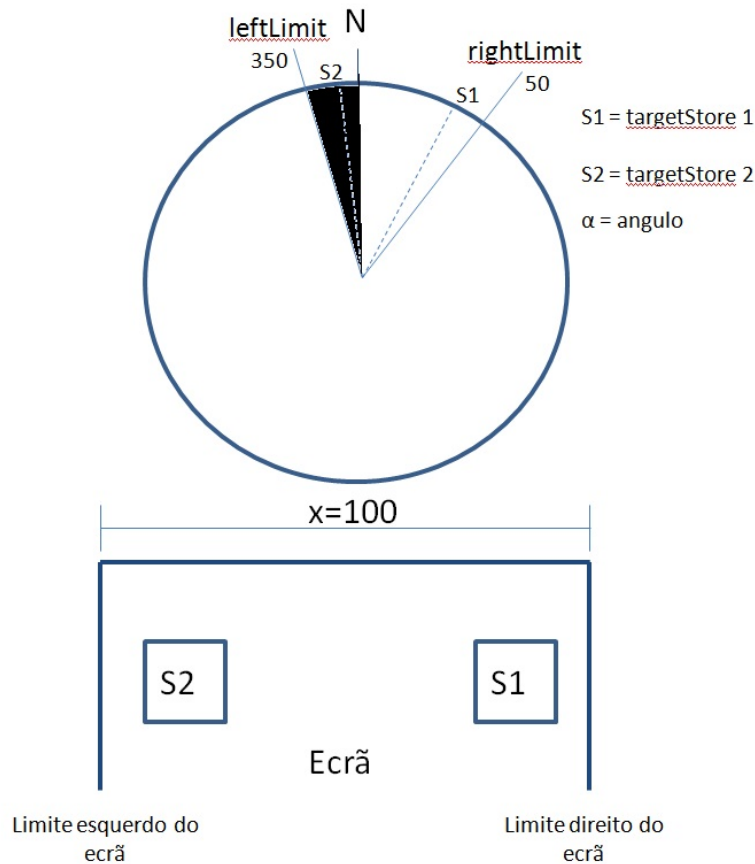


Figura 4.9: b) Ângulo da loja maior que 0 e limite direito do campo de visão menor que ângulo da loja

No caso *a)*, o procedimento é exatamente igual ao da situação 1 já explicada, e não terá nada a acrescentar.

Contudo, no caso *b)* a situação é diferente. Tendo em conta que a subtração do ângulo da store ao limite direito resulta num valor negativo, o procedimento é efectuado com base no valor do limite esquerdo.

```
1 angulo = targetStore - leftLimit;  
2 percent = angulo * 100 / (horizontalAngle * 2);  
3 x = (percent) / 100;
```

Uma vez que neste caso o cálculo é efectuado com base no limite esquerdo, a percentagem x (representativa da posição da loja no ecrã) será igual à percentagem obtida do ângulo de visão.

O resultado da situação 2 é então o seguinte:

```
1 if (targetStore > leftLimit || targetStore < rightLimit) {
```


Implementação

```
2      angulo = 0;
3      percent = 0;
4      if (rightLimit > targetStore) {
5          angulo = rightLimit - targetStore;
6          percent = angulo * 100 / (horizontalAngle * 2);
7          x = (100 - percent) / 100;
8      } else {
9          angulo = targetStore - leftLimit;
10         percent = angulo * 100 / (horizontalAngle * 2);
11         x = (percent) / 100;
12     }
13     horizontalDraw = true;
14 } else
15     horizontalDraw = false;
```

Relativamente ao posicionamento vertical da imagem das lojas no *display*, uma vez que de acordo com o estudo feito na secção 2.1.5 do capítulo 2 se concluiu ser inviável o cálculo preciso da altitude da loja em questão, então foi decidido implementar-se uma solução baseada na inclinação do dispositivo. Assim, à medida que se rodar o dispositivo para baixo (na direção que o levará a ficar paralelo ao chão) a imagem da loja é desenhada mais acima no ecrã. Vice-versa também acontece, inclinando para cima, a imagem representativa da loja desce.

Desta forma o utilizador tem de apontar em frente com o dispositivo por forma a poder efetivamente ver onde estão as lojas que o rodeiam.

4.2.1.6 Overlay de imagens

O *overlay* de imagens consiste em sobrepor a imagem produzida pela câmara com outras imagens e/ou texto criadas virtualmente.

Existem 2 formas básicas de desenhar objetos 2D em *Android*. Usar diretamente uma *View* é uma melhor solução quando o objeto é estático. Contudo, se o objeto for dinâmico ou se for um objeto que se pretende re-desenhar regularmente, então utilizar a classe *Canvas* será uma solução mais adequada.³

A facilidade de desenho recorrendo a este método foi um fator importante aquando da decisão de seguir o método *Canvas* para a implementação que resolve o problema em questão.

4.2.2 Aplicação Cardmobili

A aplicação já existente da *Cardmobili* contém informação preponderante para a implementação do módulo de realidade aumentada.

Para ser feita a comunicação entre ambos foi implementada uma *interface* que será o elo de comunicação entre os 2 módulos.

³<http://www.linux.com/learn/tutorials/707993howtodraw2dobjectinandroidwithacanvas>

Implementação

```
1 public interface AugmentedRealityInterface {
2     public void startLoyalty(String cardId);
3     public void startCoupons(Location loc);
4     public List<Store> getStores(String latitude, String longitude, String radius);
5     public Bitmap getMerchantImage(String imageId, int width, int height);
6     public int getLayoutBackgroundColor();
7     public int getLayoutTextColor();
8 }
```

Quando o utilizador toca no ecrã, é verificado se nesse ponto existe alguma loja e, no caso de existir, é aberto o respetivo programa (Fidelização ou Cupão).

```
1 if(!TextUtils.isEmpty(store.getLoyaltyCardId())) {
2     AugmentedMain.this.callback.startLoyalty(store.getLoyaltyCardId());
3 }else{
4     AugmentedMain.this.callback.startCoupons(store.getLocation());
5 }
```

Também as cores que serão atribuídas ao desenho do *overlay* no ecrã são definidas na aplicação mãe e, uma vez mais, a comunicação é feita através da interface desenvolvida.

4.2.3 Servidor

No protótipo desenvolvido é necessário saber a informação das lojas que rodeiam o utilizador e essa informação é pedida ao servidor através da aplicação mãe da *Cardmobili*.

É através da mesma interface *AugmentedRealityInterface* que a comunicação é feita por intermédio do método *getStores* e assim se preenchem, com a informação necessária, os objetos *Store* existentes no módulo de realidade aumentada.

Uma vez que esta chamada ao servidor é frequente, foi necessário recorrer a uma *AsyncTask* para executar num *thread* em *background* e posteriormente ser atualizado o ecrã com a informação pretendida.

Capítulo 5

Resultados

5.1 Primeira versão de teste

No final do período estabelecido para o desenvolvimento da dissertação foram atingidos resultados satisfatórios. O módulo de realidade aumentada foi desenvolvido com sucesso tendo atingido uma versão apresentável. Numa primeira fase foi necessário construir e testar a leitura de diversos dados como os obtidos para a localização e para os sensores. Depois foi importante fazer a integração do protótipo desenvolvido com a aplicação já existente da *Cardmobili* e aqui testar a interface desenvolvida para o efeito. Também o *overlay* de imagens sobre a câmara foi desenvolvido inicialmente numa fase de teste. A figura 5.1 ilustra uma primeira versão do protótipo.

Resultados

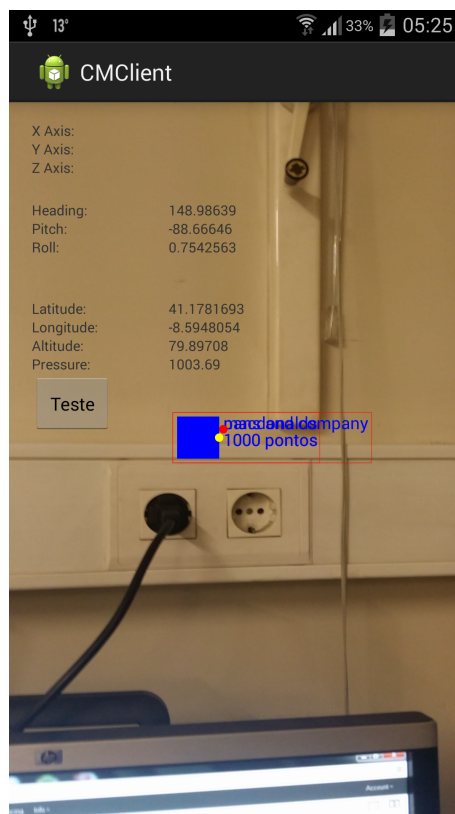


Figura 5.1: Primeira versão do protótipo

Nesta fase, no ecrã, foram apresentadas as diversas leituras de dados feitas para confirmar se os dados de localização e orientação estavam a ser corretamente adquiridos. O botão de teste, visível na figura 5.1, serviu para validar a comunicação com a aplicação mãe através da interface, uma vez que clicando no botão abriria um cartão específico. Também o *overlay* estava ainda numa fase de esboço, com valores pré definidos para o nome da loja e pontos, ainda sem a comunicação com o servidor a funcionar. Na imagem acima vemos a loja *Pans Company* com 1000 pontos. Estes valores não vieram do servidor, foram manualmente introduzidos, nesta fase, como teste.

5.2 Versão integrada na aplicação *Cardmobili*

Com a evolução da implementação, atingiram-se os resultados finais.

Resultados

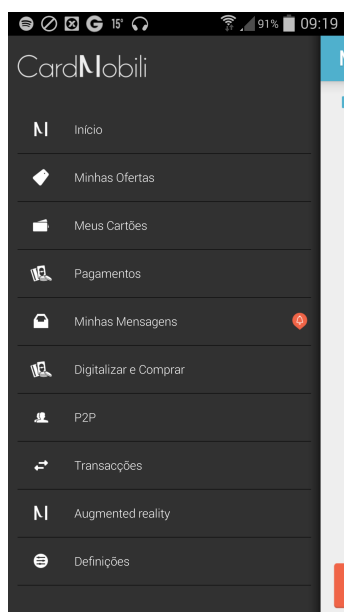


Figura 5.2: Aplicação *Cardmobili* com módulo de realidade aumentada integrado

Feita a integração do módulo de realidade aumentada com a aplicação da *Cardmobili*, o utilizador abrindo a aplicação tem a possibilidade de escolher a opção *Augmented Reality* (5.2) que iniciará o protótipo desenvolvido (figuras 5.3, 5.4, 5.5).



Figura 5.3: Versão final do protótipo

Resultados



Figura 5.4: Versão final do protótipo



Figura 5.5: Versão final do protótipo

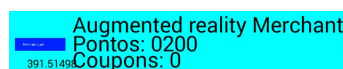


Figura 5.6: Exibição da informação da loja

Resultados

Augmented reality 2	Augmented reality M
Pontos: 0	200
Coupons: 0	0

Figura 5.7: Exibição da informação da loja

Nas figuras podem identificar-se várias lojas, pertencentes a 2 comerciantes diferentes (os de cor azul e os de cor vermelha). Nas figuras é possível observar que o utilizador possui o mesmo número de pontos em todas as lojas do comerciante azul (200 pontos) e 0 cupões, enquanto que na loja do comerciante vermelho não possui pontos nem cupões. Estas informações foram obtidas através de pedidos feitos ao servidor, processo que foi previamente explicado no capítulo 4 de implementação.

Tocado numa loja, é aberto o cupão/cartão respetivo do utilizador. No caso ilustrado na figura 5.8 foi aberto um cartão do comerciante *Pans & Company* em que o utilizador *John Smith* tinha 851 pontos na altura.

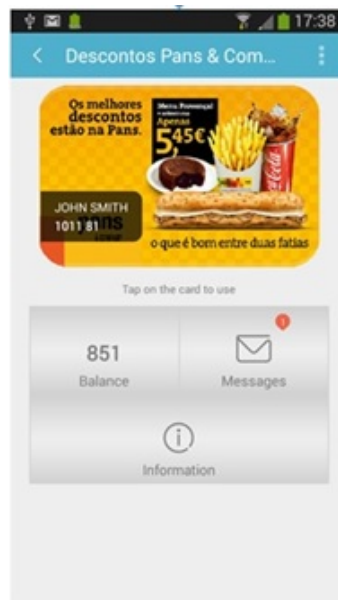


Figura 5.8: Exemplo de cartão do utilizador

Com estes resultados ficou criada uma base de desenvolvimento para uma aplicação mais complexa com base em realidade aumentada orientada a programas de ofertas e fidelização. O produto desenvolvido permite à empresa em questão oferecer visibilidade aos seus parceiros incluindo-os neste módulo, bem como oferece aos utilizadores uma nova forma de descobrir diferentes programas de fidelização e ofertas em seu redor, ou de recordar informações (como por exemplo os pontos) dos seus cartões para as diferentes lojas.

Resultados

Capítulo 6

Conclusões e trabalhos futuros

Este capítulo apresentará uma perspectiva geral do trabalho desenvolvido comentando os resultados atingidos e também o trabalho futuro que se poderá desenvolver.

6.1 Resumo e conclusões

Nesta dissertação apresentou-se o contexto do problema, a motivação para a sua resolução e os seus objetivos. Depois, apresentaram-se alguns aspetos do estado de arte, onde se fez um estudo sobre as diversas tecnologias e metodologias que podem ter interesse para a resolução do problema, detalhando cada uma delas e discutindo as vantagens e desvantagens que poderão apresentar. Documentou-se também a arquitetura proposta para o desenvolvimento do protótipo e detalhou-se na implementação as decisões tomadas nos pontos relevantes para a dissertação, explicando como e porque foram implementados.

Concluiu-se que houve uma grande evolução ao longo das versões do *Android*, nos algoritmos e processos de determinação da localização em interiores, particularmente pela *Google* que apresenta resultados bastante satisfatórios nesta questão. A precisão atingida pelos algoritmos da *Google* aliada à versatilidade que apresenta na gestão de bateria conferem um grande potencial a esta útil ferramenta de localização. Segundo o estudo feito, a precisão deste método aparenta superar os anteriores a um nível bastante satisfatório para o objetivo pretendido tanto em meios interiores como exteriores, onde o primeiro caso constituía um maior problema na generalidade dos algoritmos existentes e na implementação feita atinge uma precisão bastante satisfatória.

Concluiu-se também que o *Android* possui ferramentas simples e eficazes no que respeita à utilização dos sensores e câmara dos dispositivos, tendo sido relativamente fácil a integração destes componentes e posterior utilização.

No âmbito da realidade aumentada, notou-se ser uma área ainda em emersão, que pode ainda ser alvo de bastantes melhorias, como por exemplo bibliotecas que permitam uma mais fácil integração de imagens virtuais sobre as imagens de uma câmara por exemplo. O aparecimento de plataformas como o *AppCelerator* contribui para o desenvolvimento desta área, contudo apresenta

ainda algumas lacunas como a que foi descrita nesta dissertação e que inviabilizou o desenvolvimento do módulo através desta plataforma. Numa sociedade onde o *Android domina*, conclui-se que deve haver ainda muito trabalho a ser realizado para desenvolver a realidade aumentada.

6.2 Satisfação dos objetivos

Perante o protótipo desenvolvido o autor considera que os resultados atingidos foram satisfatórios. Conseguiu-se atingir os objetivos propostos na secção 1.4 do capítulo 1 onde o produto desenvolvido oferece os benefícios esperados que foram mencionados em 1.5, no mesmo capítulo. O autor acredita, contudo, que seria possível um protótipo mais completo, com uma melhor *interface* e resolvendo as situações em que se verifica sobreposição de lojas.

6.3 Desenvolvimentos necessários

No produto final, faltou a implementação de algumas funcionalidades.

- O raio no qual é feita a pesquisa está neste momento definido pelos algoritmos da empresa, não estando neste momento editável por parte do utilizador.
- Não está prevista a situação em que o utilizador fique sem acesso à internet durante a execução do programa. Deveria guardar em *cache* a última informação obtida das lojas e fazer esse display, bem como notificar o utilizar sobre a necessidade de ter acesso a WiFi ou Dados Móveis.
- O layout de apresentação das lojas necessita ainda de melhorias por forma a ficar mais agradável e fácil de seleccionar.
- A implementação feita nas APIs do servidor e na aplicação mãe apenas contempla cartões de fidelização dos utilizadores, não tendo sido feito o mesmo trabalho para cupões de desconto.
- O módulo não soluciona a sobreposição de lojas. Se houver mais do que uma loja no mesmo local, apenas a mais próxima é exibida no monitor e, se clicada, abre a informação dos cartões das lojas que estão por trás, por ordem.

6.4 Trabalho Futuro

Perante o desenvolvido, e após terem sido mencionados os desenvolvimentos necessários para o módulo ficar completo como versão inicial, há ainda algumas sugestões de trabalho futuro que podem ser mencionadas.

- Oferecer ao utilizador a possibilidade de mudar entre uma vista de realidade aumentada para uma vista de lista ou vista de mapa (onde em ambas seriam apresentados os vários cupões/cartões nas lojas circundantes).
- Implementação de leituras dos valores de altitude recorrendo a informações oficiais obtidas por estações meteorológicas.

Conclusões e trabalhos futuros

- Implementação de uma alternativa ao ponto anterior. Quando o utilizador entrasse num edifício, indicaria na *interface* do módulo qual o piso em que se encontra. Com essa informação, e sabendo o piso em que se encontram cada uma das lojas em particular, seria possível saber a pressão atmosférica no corrente piso e, assim medindo periodicamente a pressão atual, seria possível identificar se o utilizador subia/descia de piso, e portanto seria possível fazer a exibição da informação de lojas com uma noção de verticalidade muito boa, ainda que estivessem em pisos diferentes.
- Outra opção possível que complementaria o tópico anterior seria a indicação do piso de entrada de forma automatizada. Se houvesse um seguimento da localização com início no exterior do edifício e conhecimento prévio das entradas para o edifício numa base de dados seria então possível associar, a cada entrada, o piso a que dá acesso.
- A implementação de um radar onde fossem facilmente identificados os pontos próximos de interesse, para que o utilizador não tivesse que girar com o dispositivo para ver se existe ou não algum ponto de interesse em determinada direção.

Conclusões e trabalhos futuros

Referências

- [Bol11] Philipp Lukas Bolliger. *Robust Indoor Positioning through Adaptive Collaborative Labeling of Location Fingerprints*. PhD thesis, ETH, 2011.
- [CJJA04] Hae Don Chon, Sibum Jun, Heejae Jung e Sang Won An. Using rfid for accurate positioning. *Positioning*, 1:0, 2004.
- [Dąb11] Piotr Dąbrowski. Implementing augmented reality for visualisation of virtual buildings using android. 2011.
- [FNI13] Zahid Farid, Rosdiadee Nordin e Mahamod Ismail. Recent advances in wireless indoor localization techniques and system. *Journal of Computer Networks and Communications*, 2013, 2013.
- [GG13] Jens Grubert e Raphael Grasset. *Augmented Reality for Android application development*. Packt Publishing Ltd., November 2013.
- [GL] Julia Patzelt Gerhard Lammel. Pressure sensors provide indoor competency for navigation. Bosch SensorTech, Disponível em <http://electroiq.com/blog/2009/07/pressure-sensors-provide-indoor-competency-for-navigation/>.
- [GR08] Qing Fu Guenther Retscher. Active rfid trilateration for indoor positioning. Disponível em <http://mycoordinates.org/active-rfid-trilateration-for-indoor-positioning/all/1/>, Maio 2008.
- [JE10] Gang Xu Jay Esfandiyar, Massimo Mascotto. Mems pressure sensors in pedestrian navigation. STMicroelectronics , Disponível em <http://www.sensorsmag.com/electronics-computers/consumer/mems-pressure-sensors-pedestrian-navigation-7896>, Dezembro 2010.
- [Jia12] Landu Jiang. *A WLAN Fingerprinting Based Indoor Localization Technique*. PhD thesis, University of Nebraska, 2012.
- [Kat11] Rimma Kats. Valpak pushes geo-targeted coupons via mobile augmented reality app. Mobile Commerce Daily , Disponível em <http://www.mobilecommercedaily.com/valpak-pushes-geo-targeted-coupons-via-mobile-augmented-reality-app>, Março 2011.

REFERÊNCIAS

- [KSOK06] Masakatsu Kourogi, Nobuchika Sakata, Takashi Okuma e Takeshi Kurata. Indoor/outdoor pedestrian navigation with an embedded gps/rfid/self-contained sensor system. In *Advances in Artificial Reality and Tele-Existence*, pages 1310–1321. Springer, 2006.
- [LQD08] Binghao Li, Ishrat J Quader e Andrew G Dempster. On outdoor positioning with wi-fi. *Positioning*, 1:0, 2008.
- [LSDR06] Binghao Li, James Salter, Andrew G Dempster e Chris Rizos. Indoor positioning techniques based on wireless lan. In *LAN, First IEEE International Conference on Wireless Broadband and Ultra Wideband Communications*. Citeseer, 2006.
- [LZZ⁺09] Yin Lou, Chengyang Zhang, Yu Zheng, Xing Xie, Wei Wang e Yan Huang. Map-matching for low-sampling-rate gps trajectories. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 352–361. ACM, 2009.
- [MS12] Greg Milette e Adam Stroud. *Professional Android™ Sensor Programming*. John Wiley and Sons, Inc, 2012.
- [RD08] Lionel Reyero e Gilles Delisle. A pervasive indoor-outdoor positioning system. *Journal of Networks*, 3(8), 2008.
- [Sam13] Samsung. Developing android application using atmospheric pressure sensor. STMicroelectronics , Disponível em <http://developer.samsung.com/android/technical-docs/Developing-Android-Application-Using-Atmospheric-Pressure-Sensor/>, february 2013.
- [SK08] Nattapong Swangmuang e Prashant Krishnamurthy. Location fingerprint analyses toward efficient indoor positioning. In *Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on*, pages 100–109. IEEE, 2008.
- [Soo12] Raghav Sood. *Pro Android augmented reality*. Apress, 2012.
- [VK] Stefan Varga e Michal Kostic. Android sensors.